



TESIS - KI142502

## **Penggabungan Fitur Dimensi Fraktal dan Lacunarity untuk Klasifikasi Daun**

MUTMAINNAH MUCHTAR  
5113201004

DOSEN PEMBIMBING  
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.  
Dr. Eng. Chastine Fatichah, S.Kom., M.Kom.

PROGRAM STUDI MAGISTER  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2015



THESIS- KI142502

## **Fractal Dimension and Lacunarity Combination for Leaf Classification**

MUTMAINNAH MUCHTAR  
5113201004

SUPERVISOR  
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.  
Dr. Eng. Chastine Fatichah, S.Kom., M.Kom.

MAGISTER PROGRAM  
DEPARTMENT OF INFORMATICS ENGINEERING  
FACULTY OF INFORMATION TECHNOLOGY  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2015

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar  
Magister Komputer (M.Kom.)

di

Institut Teknologi Sepuluh Nopember Surabaya

Oleh:

MUTMAINNAH MUCHTAR

NRP. 5113201004

Dengan judul:

Penggabungan Fitur Dimensi Fraktal dan *Lacunarity* untuk Klasifikasi Daun

Tanggal Ujian : 07 Juli 2015

Periode Wisuda : September 2015

Disetujui oleh:

**Dr.Eng. Nanik Suciati, S.Kom., M.Kom.**

NIP. 197104281994122001



(Pembimbing 1)

**Dr. Eng. Chastine Fatichah, S.Kom., M.Kom.**

NIP. 197512202001122002



(Pembimbing 2)

**Dr. Agus Zainal Arifin, S.Kom M.Kom**

NIP. 197208091995121001



(Penguji 1)

**Dr. Darlis Heru Murti, S.Kom, M.Kom.**

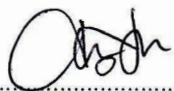
NIP. 197712172003121001



(Penguji 2)

**Anny Yuniarti, S.Kom., M.Comp.Sc**

NIP. 198106222005012002



(Penguji 3)



Direktur Program Pascasarjana,

**Prof. Dr. Ir. Ach. Soeprijanto, M.T**

NIP. 196404051990021001

## **Penggabungan Fitur Dimensi Fraktal dan *Lacunarity* untuk Klasifikasi Daun**

Nama Mahasiswa : Mutmainnah Muchtar  
NRP. : 5113201004  
Pembimbing : 1. Dr. Eng. Nanik Suciati, S.Kom., M.Kom  
2. Dr. Eng. Chastine Fatichah, S.Kom., M.Kom

### **ABSTRAK**

Tanaman memegang peranan penting dalam kehidupan manusia dan makhluk hidup lainnya. Dengan semakin tingginya keanekaragaman spesies tanaman di dunia, sulit untuk mengidentifikasi atau mengklasifikasi tanaman secara manual melalui pengamatan langsung. Perkembangan penelitian di bidang pengolahan citra digital telah membuka kesempatan luas bagi banyak peneliti di berbagai bidang penelitian untuk mengklasifikasi tanaman secara cepat dan otomatis. Daun merupakan bagian pada tanaman yang paling sering digunakan dalam klasifikasi tanaman, baik secara manual maupun otomatis. Melalui pengamatan pada daun, beberapa karakteristik bisa diperoleh; di antaranya adalah bentuk pinggiran daun, bentuk urat daun serta tekstur daun.

Banyak objek-objek di alam memiliki sifat yang mirip fraktal, dimana terdapat pola yang berulang pada skala tertentu, termasuk pada objek seperti daun. Dimensi fraktal merupakan deskriptor fitur bentuk maupun tekstur yang telah banyak diterapkan pada berbagai bidang penelitian karena mampu mendeskripsikan kompleksitas sebuah objek dalam bentuk dimensi pecahan. Sementara itu, *lacunarity*, merupakan deskriptor fitur tekstur yang mampu menunjukkan seberapa heterogen suatu citra tekstur. Namun *lacunarity* belum cukup dieksplorasi dalam banyak bidang penelitian dan belum ada penelitian signifikan yang mencoba menggabungkan fitur dimensi fraktal dengan *lacunarity* dalam penelitian yang berfokus pada klasifikasi citra digital daun.

Pada penelitian ini, diajukan penerapan konsep fraktal dalam menyelesaikan masalah klasifikasi daun dengan berfokus pada penggabungan fitur dimensi fraktal dan *lacunarity*. Ekstraksi fitur bentuk pinggiran dan tulang daun dilakukan melalui perhitungan dimensi fraktal dengan menerapkan metode *box counting*. Sedangkan fitur hasil perhitungan nilai *lacunarity* diperoleh melalui proses ekstraksi fitur tekstur daun dengan menerapkan metode *gliding box*. Menggunakan 626 dataset dari *flavia*, pengujian dilakukan dengan menganalisis performa dari dimensi fraktal dan *lacunarity* ketika digunakan secara terpisah dan ketika dikombinasikan satu sama lain dalam memperbaiki hasil klasifikasi daun dari metode fraktal sebelumnya, serta dengan mempertimbangkan parameter ukuran kotak  $r$  yang paling optimal. Hasil uji coba dengan pengklasifikasi *support vector machine* menunjukkan bahwa penggabungan fitur dimensi fraktal dan *lacunarity* mampu meningkatkan akurasi klasifikasi hingga 93.92 %

**Kata kunci:** klasifikasi daun, dimensi fraktal, *lacunarity*, *box counting*, *gliding box*

# Fractal Dimension and Lacunarity Combination for Leaf Classification

By : Mutmainnah Muchtar  
Student's ID : 5113201004  
Advisors : 1. Dr. Eng. Nanik Suciati, S.Kom., M.Kom  
2. Dr. Eng. Chastine Fatichah, S.Kom., M.Kom

## ABSTRACT

Plant plays an important role in the existence of all beings in the world. With the high diversity in plant species, it is hard to classify plant manually only by observing their properties. The development of study in digital image processing opened a wide chance for many researches from various area of study to quickly and automatically classify plant species. Plant leaf was the main properties that commonly used in plant classification whether it is manually or automatically. By looking at plant leaf, some unique characteristics can be obtained; between them were leaf contour shape, leaf vein shape, and leaf surface texture.

There are many natural objects and phenomena that have characteristic of fractals, like a pattern that repeated in a certain scale, including natural objects like plant leaf. Fractal dimension was a widely known feature descriptor for shape or texture that able to describe the complexity of an object in a form of fractional dimension's value. On the other hand, lacunarity is a feature descriptor that able to describe the heterogeneity of a texture image. However, lacunarity was not really exploited in many fields and there are no significant efforts that trying to combine fractal dimension and lacunarity in the study of automatic plant leaf classification.

In this study, a fractal concept and its performances in leaf classification will be analyzed by using two fractal based feature: fractal dimension and lacunarity. We focused on how to extract the two features and combine them for a better classification result. A box counting approach is implemented to get the fractal dimension feature vectors of leaf contour and vein, while an improved gliding box algorithm is implemented to get the lacunarity feature vectors of leaf texture. By combining this two feature, a feature vectors that highly represents the unique feature of each leaf is then expected to be obtained. Using 626 leaf images from *flavia*, experiment was conducted by separately or jointly analyzing the performance of both fractal dimension feature vectors and lacunarity feature vectors, while considering the optimal box size  $r$ . Using support vector machine classifier, result shows that combination between fractal dimension and lacunarity was able to increase the classification accuracy up to 93.92%.

**Keywords:** *leaf classification, fractal dimension, lacunarity, box counting, gliding box*

## KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Puji syukur kita panjatkan ke hadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga buku tesis ini dapat diselesaikan dengan baik. Meski dalam menyelesaikan buku ini banyak ditemui kesulitan, namun berkat bantuan dan bimbingan berbagai pihak, akhirnya penulis berhasil menyelesaikan buku ini. Untuk itu atas segala bantuan yang telah diberikan, penulis ingin mengucapkan terima kasih serta penghargaan yang sebesar-besarnya kepada:

1. Kedua Orang tua saya, Bapak Muchtar Ibrahim dan Ibu Nurhayati S. atas kasih sayang yang tiada bandingannya, Adik-adik saya Mardhiyah dan Rafiqah, Kakak saya Amsal, dan seluruh keluarga besar atas dukungan dan doanya sehingga saya bisa mendapatkan kemudahan dalam menyelesaikan studi ini. Semoga Allah SWT senantiasa memberikan rahmat, kesehatan, dan keselamatan kepada kita semua.
2. Ibu Dr.Eng. Nanik Suciati, S.Kom, M.Kom. dan Ibu Dr.Eng. Chastine Fatichah, S.Kom., M.Kom., selaku Dosen Pembimbing I dan II yang telah banyak meluangkan waktu dan pikiran sampai terselesaikannya tesis ini. Semoga Allah SWT senantiasa merahmati ibu dan keluarga.
3. Bapak Dr. Agus Zainal Arifin, S.Kom M.Kom, Bapak Dr. Darlis Heru Murti, S.Kom, M.Kom, dan Ibu Anny Yuniarti, S.Kom., M.Comp.Sc selaku dosen penguji yang telah banyak membantu saya untuk bisa menjadi lebih baik.
4. Seluruh dosen S2 Teknik Informatika ITS yang telah memberikan wawasan serta ilmu pengetahuan baru bagi saya selama menempuh masa studi pascasarjana.
5. Sahabat-sahabat saya, Anak Sepatu, yang telah bersama berjuang di kampus perjuangan: Lala, Yayan, Mba Yuyu, dan Ichal. Terima kasih atas segala suka dan duka yang telah dibagi selama masa studi di kota pahlawan ini. Semoga ilmu yang kita peroleh bisa membawa kebaikan bagi kita dan orang lain. Juga kepada sahabat saya Anggit dan Fitri atas semua kenangan pada saat tes

masuk kampus ITS ini. Semoga persaudaraan kita terus berlanjut dan kita tetap dalam perjuangan.

6. Teman seperjuangan saya, Mbak Laili, serta teman-teman seangkatan yang telah berbagi dan saling menyemangati dalam masa masa perkuliahan hingga masa penulisan tesis. Semoga ilmu yang kita dapatkan bisa bermanfaat dan penuh berkah.
7. Kepada semua sahabat-sahabat saya di Kendari atas dukungan morilnya, juga kepada teman-teman dan pihak-pihak lain yang tidak bisa saya sebutkan satu-persatu, terima kasih atas segala bantuan, baik berupa ide, gagasan, dan pemikiran yang diberikan selama saya menempuh studi.

Tak ada gading yang tak retak. Saya menyadari bahwa dalam buku tesis ini masih terdapat banyak kekurangan. Oleh karena karena itu, masukan, saran, serta kritik yang bersifat membangun demi perbaikan dan pengembangan tesis ini sangat saya harapkan. Semoga tesis ini dapat bermanfaat bagi saya, masyarakat, bangsa dan negara.

Surabaya,        Juli 2015

Penulis

## DAFTAR ISI

<b>ABSTRAK .....</b>	<b>VII</b>
<b>ABSTRACT .....</b>	<b>IX</b>
<b>KATA PENGANTAR.....</b>	<b>XI</b>
<b>DAFTAR ISI.....</b>	<b>XIII</b>
<b>DAFTAR GAMBAR.....</b>	<b>XV</b>
<b>DAFTAR TABEL .....</b>	<b>XVII</b>
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Tujuan Penelitian .....	4
1.4 Manfaat Penelitian .....	4
1.5 Kontribusi Penelitian.....	4
1.6 Batasan Masalah .....	4
<b>BAB 2 KAJIAN PUSTAKA .....</b>	<b>5</b>
2.1 Klasifikasi Tanaman Melalui Daun .....	5
2.2 Konsep Fraktal.....	7
2.2.1 Dimensi Fraktal sebagai Deskriptor Bentuk.....	8
2.2.2 <i>Lacunarity</i> sebagai Deskriptor Tekstur .....	10
2.3 Deteksi Tepi.....	13
2.3.1 Operator Deteksi Tepi <i>Canny</i> .....	13
2.3.2 Operator Deteksi Tepi <i>Kirsch</i> .....	14
2.4 <i>k-fold Cross Validation System</i> .....	14
2.5 <i>Support Vector Machine</i> .....	15
<b>BAB 3 METODOLOGI PENELITIAN.....</b>	<b>17</b>
3.1 Analisis dan Perancangan Sistem .....	17
3.1.1 Deskripsi <i>Dataset</i> .....	18
3.1.2 Praproses dan Segmentasi .....	18
3.1.3 Ekstraksi Fitur Dimensi Fraktal dan <i>Lacunarity</i> .....	23
3.1.4 Klasifikasi.....	25



3.2 Implementasi.....	25
3.3 Uji Coba dan Evaluasi.....	25
3.4 Analisis Hasil.....	27
<b>BAB 4 HASIL DAN PEMBAHASAN.....</b>	<b>29</b>
4.1 Lingkungan Uji Coba.....	29
4.2 Data Uji Coba.....	29
4.3 Praproses.....	30
4.4 Segmentasi Pinggiran dan Urat Daun.....	31
4.5 Ekstraksi Fitur.....	32
4.5.1 Ekstraksi Fitur Dimensi Fraktal.....	32
4.5.2 Ekstraksi Fitur <i>Lacunarity</i> .....	35
4.6 Uji Coba.....	36
4.6.1 Hasil Uji Coba Fitur Dimensi Fraktal.....	37
4.6.2 Hasil Uji Coba Fitur <i>Lacunarity</i> .....	39
4.6.3 Hasil Uji Coba Fitur Gabungan Dimensi Fraktal dan <i>Lacunarity</i> .....	41
4.7 Pembahasan Hasil Uji Coba.....	43
4.7.1 Analisis Uji Coba Fitur Dimensi Fraktal.....	43
4.7.2 Analisis Uji Coba Fitur <i>Lacunarity</i> .....	44
4.7.3 Analisis Uji Coba Penggabungan Fitur Dimensi Fraktal dan <i>Lacunarity</i> ....	47
<b>BAB 5 KESIMPULAN DAN SARAN.....</b>	<b>49</b>
5.1. Kesimpulan.....	49
5.2 Saran.....	50
<b>DAFTAR PUSTAKA.....</b>	<b>51</b>
<b>BIODATA PENULIS.....</b>	<b>56</b>

## DAFTAR TABEL

Tabel 2.1	Perhitungan Nilai <i>Lacunarity</i> Dari Citra Masukan .....	11
Tabel 3.1	Aspek Rasio <i>R</i> dengan Ukuran Citra Berbeda .....	20
Tabel 3.2	Desain Data Pengujian.....	26
Tabel 4.1	Keterangan <i>Dataset</i> .....	29
Tabel 4.2	Contoh Proses Perhitungan Nilai Fitur Dimensi Fraktal dari Pinggiran Daun Melalui Grafik dengan Metode <i>Box Counting</i> .....	33
Tabel 4.3	Contoh Proses Perhitungan Nilai Fitur Dimensi Fraktal dari Urat Daun Melalui Grafik dengan Metode <i>Box Counting</i> . ....	34
Tabel 4.4	Hasil Perhitungan Nilai <i>Lacunarity AR</i> dari 6 Citra Masukan .....	36
Tabel 4.5	Perbandingan Hasil Klasifikasi Antara Citra Yang Mengalami Rotasi dan <i>Shrinking</i> Dengan Citra Tanpa Rotasi dan <i>Shrinking</i> . ....	38
Tabel 4.6	Hasil Klasifikasi Dengan Menggunakan Beragam Nilai Sigma Pada Operator <i>Canny</i> . ....	38
Tabel 4.7	Akurasi Klasifikasi Dengan Metode <i>Gliding Box Lacunarity</i> Dengan Menerapkan Metode Backes (2013).....	40
Tabel 4.8	Akurasi Klasifikasi Dengan Fitur <i>Lacunarity</i> Untuk Tiap Citra Masukan.....	40
Tabel 4.9	Akurasi Klasifikasi Dengan Menggunakan Berbagai Ukuran Kotak <i>r</i> Pada <i>Lacunarity</i> . ....	41
Tabel 4.10	Perbandingan Akurasi Klasifikasi Fitur Dimensi Fraktal, Fitur <i>Lacunarity</i> dan Fitur Gabungan. ....	42
Tabel 4.11	Citra Kelas Berbeda dengan Nilai <i>Lacunarity</i> yang Mirip .....	45

## DAFTAR GAMBAR

Gambar 2.1 (a) Bentuk-bentuk dari daun; (b) Bentuk tulang/urat daun secara umum (Dirr, 2009).....	6
Gambar 2.2 Tekstur daun merupakan pola yang ada pada permukaan daun.....	7
Gambar 2.3 Dimensi <i>Euclidean</i> yang mendefinisikan dimensi objek pada gambar: (a) titik = 0 , (b) garis = 1, (c) bidang = 2, (d) ruang = 3, namun tidak mendefinisikan dimensi objek seperti pada gambar (e). (Putra, 2010).....	8
Gambar 2.4 Kotak berukuran $r$ yang diterapkan pada citra biner 12x12 piksel (Plotnick dkk.,1993). ....	10
Gambar 2.5 Proses konversi citra <i>grayscale</i> dengan <i>thresholding</i> lokal. (Backes, 2013).....	12
Gambar 2.6 Penentuan garis pemisah (a); dan garis pemisah terbaik (b). (Nugroho dkk., 2003). ....	15
Gambar 3.1 Diagram rancangan sistem secara umum.....	17
Gambar 3.2 Contoh <i>dataset</i> dari <i>flavia</i> . ....	18
Gambar 3.3 Praproses dan segmentasi pinggiran dan urat daun.....	19
Gambar 3.4 Proses segmentasi urat daun dengan $n$ nilai ambang.....	21
Gambar 3.5 Tahapan praproses citra tekstur.....	22
Gambar 3.6 Diagram ekstraksi fitur dimensi fraktal pinggiran dan urat daun.....	23
Gambar 3.7 Diagram ekstraksi fitur <i>lacunarity</i> dari tekstur daun.....	24
Gambar 4.1 Tahap praproses citra.....	30
Gambar 4.2 Tahap praproses untuk mendapatkan citra tekstur daun.....	31
Gambar 4.3 Proses segmentasi pinggiran dan urat daun. ....	32
Gambar 4.4 Proses penentuan dimensi fraktal dari pinggiran daun. ....	33
Gambar 4.5 Proses penentuan dimensi fraktal dari urat daun.....	34
Gambar 4.6 Contoh citra hasil <i>local thresholding</i> (Backes, 2013). ....	35
Gambar 4.7 Proses rotasi dan <i>shrinking</i> citra.....	37
Gambar 4.8 Proses perhitungan dimensi fraktal dengan menerapkan 4 nilai sigma pada operator <i>canny</i> .....	39

Gambar 4.9 Grafik pengaruh nilai sigma terhadap akurasi klasifikasi dengan fitur dimensi fraktal. ....	43
Gambar 4.10 Klasifikasi dengan menerapkan metode Backes (2013). ....	44
Gambar 4.11 Tingginya similaritas antara beberapa kelas dalam <i>dataset</i> . ....	44
Gambar 4.12 Pengaruh ukuran kotak $r$ terhadap hasil klasifikasi. ....	46
Gambar 4.13 Perbandingan hasil klasifikasi fitur dimensi fraktal, fitur <i>lacunarity</i> , dan fitur gabungan.....	48

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Ilmu sains yang berkaitan dengan pengenalan atau klasifikasi tanaman memegang peranan penting dalam kehidupan manusia, termasuk dalam bidang makanan, pengobatan, industri, pertanian, dan lingkungan. Dengan semakin banyaknya spesies tanaman di dunia, penting untuk melindungi tanaman atau mengumpulkannya dalam bentuk informasi yang menawarkan keanekaragaman tanaman. Klasifikasi berbasis komputer yang mampu mengenali tanaman tentu sangat membantu para peneliti di bidang pertanian dan perkebunan, *botanist*, ahli tanaman herbal, dokter, bahkan dapat juga digunakan sebagai media pembelajaran di sekolah. Klasifikasi tanaman baik secara manual maupun otomatis dapat dilakukan melalui pengamatan pada ciri-ciri fisik pada tanaman, yaitu pengamatan pada bagian bunga, daun, buah, akar atau batang (Backes dkk., 2009). Melihat jumlahnya yang melimpah dan proses pengumpulannya yang cukup mudah, daun merupakan bagian tanaman yang paling sering digunakan dalam klasifikasi tanaman (Novotny dan Suk, 2013).

Daun memiliki banyak ciri khusus yang bisa digunakan sebagai fitur dalam proses klasifikasi, antara lain yaitu fitur warna, bentuk, tekstur, urat daun, atau kombinasi dari fitur-fitur tersebut (Beghin, 2010 dan Kadir dkk., 2011). Penelitian berbasis warna daun masih memiliki kendala, sebab sebagian besar daun memiliki warna yang sama dan beberapa jenis daun mengalami perubahan warna pada musim tertentu. Oleh karena itu, tekstur dan bentuk merupakan fitur daun yang paling sering diteliti. Contoh fitur bentuk daun telah dianalisis pada beberapa penelitian adalah seperti deskriptor fitur geometrik (Du dkk., 2007) dan fitur dimensi fraktal (Bruno dkk., 2008 dan Du dkk, 2013). Contoh penelitian yang terkait dengan pengenalan daun berbasis tekstur adalah penelitian dengan metode *Gray Level Co-Occurrence Matrix (GLCM)* dan *Local Binary Pattern (LBP)* (Arun dkk., 2013), Gabor (Lin, 2008) dan juga Fraktal (Xu, 2011). Salah

satu sifat fraktal adalah *self similarity*, dimana terjadi pola perulangan pada skala tertentu. Pola perulangan ini dapat ditemukan pada berbagai objek di alam semesta, termasuk pada daun. Oleh karena itu, penelitian ini menerapkan metode berbasis konsep fraktal dalam melakukan klasifikasi daun.

Penerapan dari konsep fraktal untuk objek yang bersifat fraktal maupun nonfraktal telah umum digunakan dalam analisis citra dan pengenalan pola, dimana dimensi fraktal dimanfaatkan dalam mengukur kompleksitas dari bentuk geometris maupun tekstur pada objek (Mandelbrot, 1982; Allain, 1991; Dong, 2000). Namun, terdapat kemungkinan bahwa objek yang memiliki pola berbeda akan menampilkan nilai dimensi fraktal yang sama. Mandelbrot (1982) kemudian memperkenalkan konsep *lacunarity* yang mampu mengukur distribusi spasial dari *gap* (celah) dari ukuran tertentu pada tekstur citra (Backes, 2013). Nilai *lacunarity* yang rendah menunjukkan bahwa tekstur bersifat homogen karena semua *gap* menunjukkan ukuran yang sama. Sedangkan *lacunarity* yang tinggi menunjukkan bahwa tekstur bersifat heterogen. *Lacunarity* telah diaplikasikan dalam beberapa bidang penelitian berbasis tekstur, seperti pada bidang pemetaan data spasial (Mynt dan Lam, 2005; Dong, 2009), medis (Boryz dkk., 2008; Neves dkk., 2014), dan industri pertanian (Li dkk., 2012).

Metode yang paling umum digunakan dalam menghitung nilai dimensi fraktal adalah metode *box counting* oleh Voss (1986), karena kemampuannya dalam merepresentasikan kerumitan dari citra serta implementasinya yang mudah (Kilic dan Abiyev, 2011). Oleh karena itu, Bruno dkk. (2008) mengidentifikasi tanaman berdasarkan kompleksitas dari bentuk internal dan eksternal dari daun untuk memperoleh fitur dimensi fraktal menggunakan metode *box counting*. Hasil penelitian tersebut menunjukkan bahwa fitur dimensi fraktal dapat memberikan performa yang cukup baik dalam masalah klasifikasi daun. Namun, *misclassification rate* pada hasil klasifikasi masih cukup tinggi, sehingga perlu dilakukan kombinasi antara fitur dimensi fraktal yang digunakan pada penelitian tersebut dengan fitur tekstur berbasis fraktal seperti *lacunarity* untuk meningkatkan akurasi.

Salah satu metode yang dikembangkan untuk mendapatkan fitur *lacunarity* adalah *gliding box* oleh Plotnick dkk. (1993). *Gliding box* adalah kotak dengan

ukuran tertentu yang diterapkan pada citra *grayscale* atau biner dari arah kiri ke kanan. Metode ini memiliki kelemahan karena proses *thresholding* citra yang bersifat global. Salah satu pengembangan metode *gliding box* yang diajukan oleh Backes dkk. (2013) adalah penerapan pola biner lokal pada citra masukan dengan *thresholding* lokal yang dilakukan pada tiap kotak. Namun, pada penelitian tersebut nilai *thresholding* ditentukan hanya melalui rata-rata nilai keabuan, sehingga penelitian ini mencoba untuk menerapkan metode *thresholding* yang lebih berkembang seperti metode *thresholding* oleh Otsu (1972). Mengingat pentingnya pola biner dalam meningkatkan sifat diskriminatif dari *lacunarity*, penerapan *thresholding* pada tiap kotak di *gliding box* diharapkan mampu mempertahankan informasi lokal tekstur yang hilang ketika menerapkan *thresholding* secara global.

Ketika dua citra memiliki pola perulangan yang sama, maka dimensi fraktal belum mampu membedakan kedua citra tersebut, meskipun keduanya memiliki penampakan visual yang berbeda. *Lacunarity* sebagai deskriptor tekstur memiliki kemampuan dalam mendeskripsikan sebaran *gap* pada citra dan menganalisis seberapa heterogen permukaan sebuah citra, sehingga bisa mengatasi kelemahan dari dimensi fraktal tersebut. Namun, menurut Kilic dan Abiyev (2011), dimensi fraktal dan *lacunarity* selama ini diteliti secara terpisah, dan belum ada usaha yang signifikan dalam menggabungkan keduanya dalam kombinasi yang lebih baik.

Oleh karena itu, penelitian ini mengusulkan penggabungan fitur dimensi fraktal dari bentuk daun dan fitur *lacunarity* dari tekstur daun dengan menggunakan pengklasifikasi *Support Vector Machine (SVM)*. *SVM* dipilih karena kelebihanannya dalam hal generalisasi. Untuk memperoleh fitur dimensi fraktal, digunakan metode *box-counting* yang diterapkan oleh Bruno dkk. (2008). Fitur *lacunarity* diperoleh dengan menggunakan metode *gliding-box* oleh Backes (2013) dengan mengajukan penerapan pola biner lokal serta penerapan ukuran *box* yang bervariasi pada perhitungan *lacunarity*. Dengan kombinasi ini, diharapkan akan terbentuk sebuah representasi fitur yang dapat meningkatkan hasil akurasi klasifikasi daun.

## **1.2 Rumusan Masalah**

Rumusan masalah dari penelitian ini diuraikan sebagai berikut:

1. Mengekstraksi fitur bentuk dimensi fraktal dan fitur tekstur *lacunarity* pada daun.
2. Menggabungkan fitur bentuk dimensi fraktal dan fitur tekstur *lacunarity* dalam proses klasifikasi daun.

## **1.3 Tujuan Penelitian**

Tujuan yang akan dicapai dalam penelitian ini adalah melakukan klasifikasi daun menggunakan penggabungan fitur bentuk berupa dimensi fraktal dan fitur tekstur berupa *lacunarity*.

## **1.4 Manfaat Penelitian**

Manfaat dari penelitian ini adalah memberikan referensi berupa kombinasi fitur bentuk dan tekstur yang tepat untuk digunakan dalam permasalahan klasifikasi daun.

## **1.5 Kontribusi Penelitian**

Kontribusi penelitian ini adalah menggabungkan fitur dimensi fraktal dari bentuk daun dan fitur *lacunarity* dari tekstur daun untuk meningkatkan akurasi pada klasifikasi daun dengan menerapkan metode *box counting* dan *gliding box*.

## **1.6 Batasan Masalah**

Batasan masalah dari penelitian ini adalah sebagai berikut:

1. Dataset yang digunakan adalah citra daun yang diunduh di *flavia leaf dataset*.
2. Implementasi sistem dilakukan dengan menggunakan perangkat lunak Matlab R2011b.



## **BAB 2**

### **KAJIAN PUSTAKA**

Pada bab ini dibahas dasar dari teori yang menjadi acuan dalam penelitian ini. Kajian pustaka yang dijelaskan meliputi klasifikasi tanaman melalui daun dan konsep fraktal. Selain itu dibahas pula fitur dimensi fraktal dan *lacunarity* serta metode untuk memperoleh fitur tersebut.

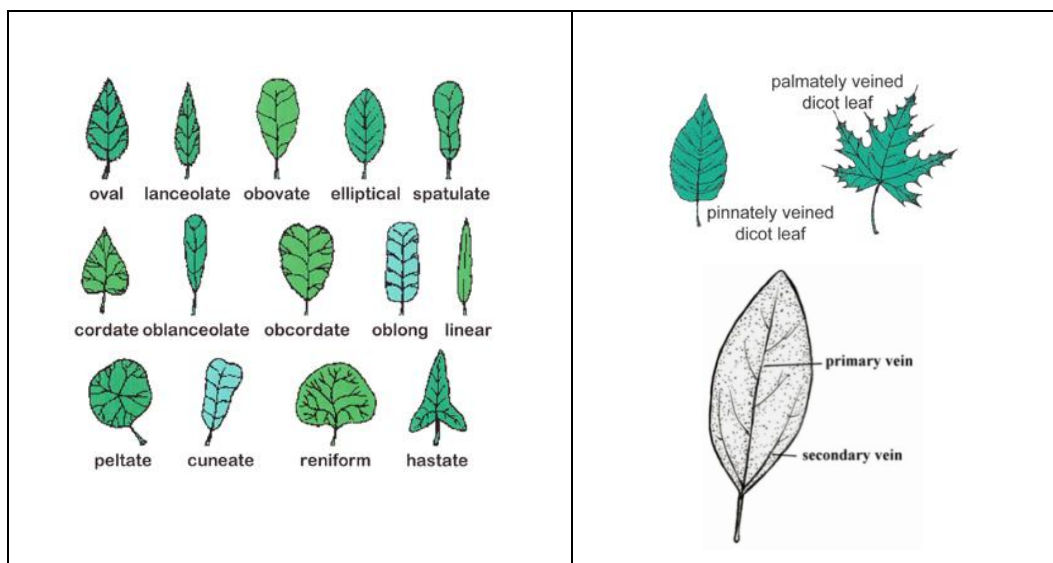
#### **2.1 Klasifikasi Tanaman Melalui Daun**

Ilmu sains yang berkaitan dengan klasifikasi tanaman memegang peranan penting dalam kehidupan manusia, termasuk dalam bidang makanan, pengobatan, industri, pertanian, biologi, dan lain sebagainya (Du dkk., 2007; Backes dkk., 2009; 2013; Arun dkk., 2013). Akan tetapi, mengidentifikasi spesies dari suatu tanaman merupakan pekerjaan yang membutuhkan waktu cukup lama dan bergantung pada tingkat keahlian tiap individu. Seseorang yang tidak memiliki pengetahuan memadai di bidang *taxonomy* bisa menghabiskan waktu hingga berjam-jam untuk mengidentifikasi suatu tanaman. Sebagai contoh, seorang mahasiswa Biologi biasanya diberikan waktu selama dua jam untuk mengidentifikasi tanaman secara manual (Sosa dkk., 2013).

*Plant Morphology* merupakan ilmu yang meneliti bagian-bagian berbeda pada tanaman melalui karakteristik fisik yang nampak dari luar (Kaplan, 2001). Karakteristik khusus tersebut dilakukan melalui pengamatan pada bunga, biji, daun dan bagian batang (Backes dkk., 2009). Di antara karakteristik tersebut, daun merupakan bagian yang paling sering digunakan dalam identifikasi tanaman (Novotny dan Suk, 2013). Beberapa metode dalam mengenali *anatomy* atau ciri fisik daun adalah melalui pengamatan pada bentuk daun secara umum, bentuk tulang atau urat daun serta melalui tekstur daun. Untuk mengenali daun secara manual, biasanya ahli akan memperhatikan detail pada bentuk umum dari daun, serta bentuk dari puncak dan pangkal daun (Dirr, 2009). Secara umum, bentuk daun dibedakan dalam beberapa jenis, seperti yang terlihat pada gambar 2.1 (a)

Selain melalui bentuk pinggirannya, identifikasi daun juga bisa dilakukan melalui pengamatan pada urat daun atau tulang daun (*leaf venation*). Urat daun

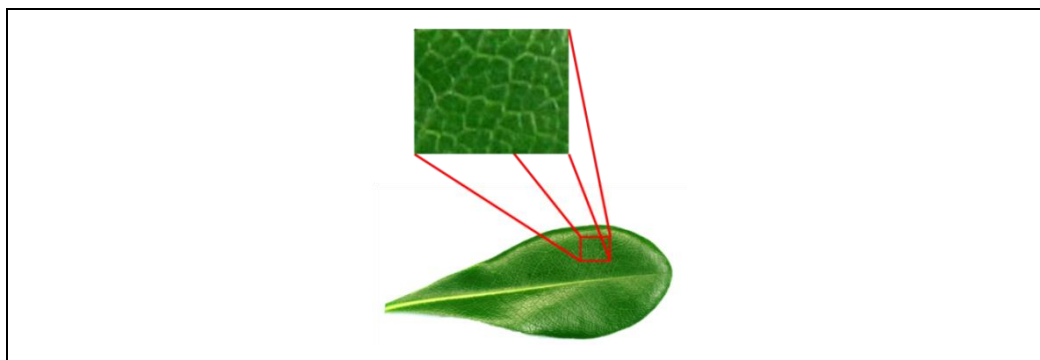
merujuk pada pola dari urat daun. Pola dari urat daun bisa digunakan dalam mengidentifikasi tanaman ketika pola tersebut nampak dengan cukup jelas. Pola dari urat daun terdiri atas dua macam yaitu: (1) urat primer (*primary vein*) adalah seperti batang pada pohon, merupakan urat terbesar pada daun dan dimulai pada bagian bawah daun; dan (2) urat sekunder (*secondary vein*), berbentuk seperti ranting utama pada pohon, dan lebih kecil daripada *primary vein*. Gambar 2.1 (b) menunjukkan bentuk umum dari urat daun.



Gambar 2.1 (a) Bentuk-bentuk dari daun; (b) Bentuk tulang/urat daun secara umum (Dirr, 2009).

Selain melalui bentuk daun, tekstur daun juga menjadi ciri khusus dalam mengidentifikasi tanaman. Dalam analisis daun secara manual, tekstur daun diidentifikasi dengan sifat permukaan dari daun (kasar, halus, licin, dan sebagainya). Dengan bantuan komputer, analisis pada citra tekstur daun bisa dilakukan lebih cepat. Meskipun belum ada definisi yang pasti mengenai tekstur pada citra digital (Gonzales dan Woods, 2002), tekstur berkaitan dengan pengukuran sifat-sifat seperti kehalusan (*smoothness*), kekasaran (*coarseness*), dan keteraturan (*regularity*) pada citra. Oleh karena itu, tekstur dari citra digital daun dapat diidentifikasi dengan area pada permukaan daun yang menunjukkan keteraturan pola-pola tertentu (Backes dkk., 2009; Casanova dkk., 2009). Pada

penelitian ini, tekstur daun yang akan dianalisis adalah area pada permukaan daun yang bukan merupakan urat daun utama, seperti terlihat pada Gambar 2.2.



Gambar 2.2 Tekstur daun merupakan pola yang ada pada permukaan daun.

Dengan semakin besarnya keberagaman flora saat ini, klasifikasi tanaman berdasarkan bentuk pinggiran dan urat daun yang memiliki kemiripan tinggi seperti pada Gambar 2.1 tentunya merupakan hal yang sangat sulit dilakukan, terutama bagi orang awam. Oleh karena itu, klasifikasi daun otomatis berdasarkan bentuk pinggiran dan urat daun maupun melalui tekstur daun merupakan hal yang sangat dibutuhkan. Klasifikasi berbasis komputer tersebut tentu sangat membantu para peneliti di bidang pertanian dan perkebunan, ahli tanaman herbal, dokter, dan lain sebagainya. Properti pada citra daun memiliki karakteristik yang mirip dengan sifat fraktal (Du dkk., 2007; Bruno dkk., 2008; Du dkk., 2013). Dengan mempertimbangkan geometri fraktal sebagai salah satu metode analisis bentuk dan tekstur citra yang telah banyak diterapkan pada penelitian sebelumnya, maka konsep fraktal dapat digunakan dalam ekstraksi fitur bentuk dan tekstur daun.

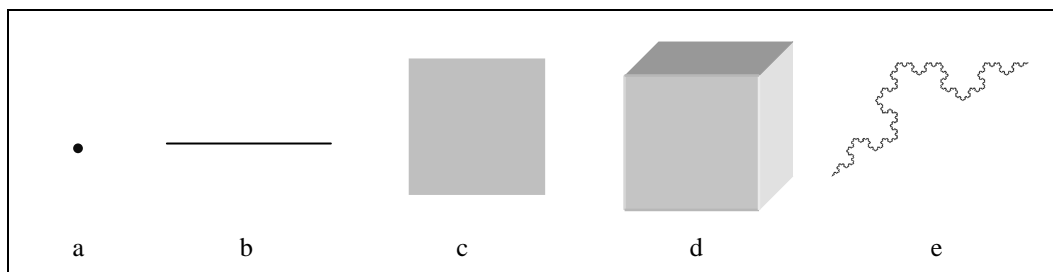
## 2.2 Konsep Fraktal

Secara harfiah, Fraktal (*fractal*) berasal dari bahasa latin yaitu *fractus* yang berarti pecah (*broken*) atau tidak teratur (*irregular*). Fraktal pertama kali diperkenalkan oleh Benoit B. Mandelbrot dalam bukunya *The Fractal Geometry of Nature* (Mandelbort, 1982). Fraktal merupakan cara yang tepat untuk membuat pemodelan fenomena alam (*natural phenomenon*) dan dianggap sebagai alat yang andal dalam merepresentasikan objek-objek alam (Putra, 2010). Banyak objek

pada alam semesta ini memiliki model yang rumit dan tidak teratur (Du dkk., 2013). Sebagai contoh, pepohonan, aliran sungai yang berliku-liku, jaringan pembuluh darah, syaraf pada retina, garis-garis telapak tangan, dan sebagainya. Seringkali objek-objek tersebut merupakan perulangan pola-pola pada berbagai skala, tidak masalah sekecil apapun ukuran skalanya. Daun juga memiliki pola-pola perulangan tertentu yang dapat dianalisis dengan metode fraktal, namun dalam skala yang terbatas.

### 2.2.1 Dimensi Fraktal sebagai Deskriptor Bentuk

Fraktal memiliki karakteristik utama yaitu kemiripan dengan diri sendiri (*self similarity*). Karakteristik tersebut membuat fraktal memiliki kemampuan memodelkan objek alam yang rumit dan tidak teratur. Menggunakan karakteristik tersebut pula fraktal mampu menentukan dimensi suatu objek. Tidak seperti geometri *Euclidean* yang hanya mampu menentukan dimensi bulat (*integer*) dari suatu objek, fraktal mampu menghasilkan dimensi pecahan (*fractional dimension*) dari suatu objek.



Gambar 2.3 Dimensi *Euclidean* yang mendefinisikan dimensi objek pada gambar: (a) titik = 0 , (b) garis = 1, (c) bidang = 2, (d) ruang = 3, namun tidak mendefinisikan dimensi objek seperti pada gambar (e). (Putra, 2010).

Seperti yang terlihat pada Gambar 2.3, geometri *Euclidean* mendeskripsikan bahwa titik (*point*) tidak memiliki dimensi karena tidak memiliki panjang, lebar maupun bobot. Garis (*line*) berdimensi satu karena hanya memiliki panjang, Bidang (*plane*) berdimensi dua karena memiliki panjang dan lebar. Ruang (*space*) berdimensi tiga karena memiliki panjang, lebar dan tinggi. Dimensi dari objek-objek *euclidean* di atas merupakan bilangan bulat. Akan tetapi, jika

diberikan citra yang tidak beraturan seperti Gambar 2.3 (e), seharusnya dimensi citra tersebut bernilai di antara 1 dan 2 karena bukan merupakan garis maupun bidang yang sempurna. Geometri fraktal mampu menghitung dimensi objek seperti pada gambar dalam bentuk pecahan. Fraktal mampu menyatakan bahwa suatu objek memiliki dimensi 1.2, 1.45, 2.04 dan lain sebagainya (Putra, 2010). Lebih lanjut lagi, dimensi fraktal mendeskripsikan ketidak-beraturan (*irregularity*) dan seberapa banyak spasi/ruang yang ditempati oleh suatu objek. (Backes dkk., 2009).

Metode yang paling sering digunakan untuk menghitung dimensi fraktal suatu citra adalah metode Penghitungan Kotak (*Box Counting*) (Voss, 1986; Bruno dkk., 2008). Dimensi fraktal suatu citra dengan metode ini dihitung dengan rumus sebagai berikut:

$$D(s) = \frac{\log (N(s))}{\log (s)}, \quad (2.1)$$

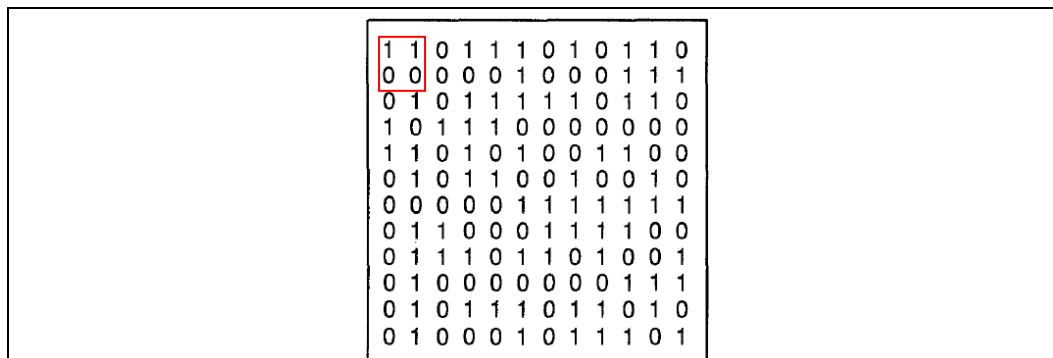
dengan  $N(s)$  menyatakan banyaknya kotak berukuran  $s$  yang berisi informasi (piksel) objek dan  $D(s)$  adalah dimensi fraktal objek dengan kotak berukuran  $s$ . Adapun langkah-langkah metode penghitungan kotak adalah sebagai berikut (Putra, 2010):

- a) Citra dibagi kedalam kotak-kotak dengan ukuran  $s$ . Nilai  $s$  berubah dari 1 sampai  $2^k$ , dengan  $k = 0, 1, 2, \dots$  dan seterusnya.  $2^k$  tidak boleh lebih besar dari ukuran citra. Bila citra berukuran  $2^m \times 2^m$ , maka nilai  $k$  akan berhenti sampai  $m$ .
- b) Hitung banyaknya kotak  $N(s)$  yang berisi bagian objek pada citra. Nilai  $N(s)$  sangat tergantung pada  $s$ .
- c) Hitung  $D(s)$  dengan persamaan (2.1) untuk seluruh nilai  $s$ .
- d) Buat garis lurus berdasarkan nilai  $\log(N(s))$  (sumbu  $y$ ) dan nilai-nilai  $\log(s)$  (sumbu  $x$ ) untuk setiap nilai  $s$ , kemudian hitung kemiringan (*slope*) dari garis lurus. Nilai dari *slope* inilah yang merupakan dimensi fraktal dari suatu citra. *Slope* dari suatu garis lurus dapat dihitung dengan metode *least square* (Putra, 2010).

### 2.2.2 *Lacunarity* sebagai Deskriptor Tekstur

*Lacunarity* berasal dari bahasa Latin (*lacuna*) yang juga merupakan asal kata *lake* dalam bahasa Inggris, merujuk pada konsep yang juga diperkenalkan oleh “bapak” dari konsep fraktal, Mandelbrot, pada tahun 1982. Konsep ini mendefinisikan bahwa suatu objek akan bersifat “*lacunar*” jika *gap* (lubang) pada objek cenderung besar. *Lacunarity* yang rendah menunjukkan bahwa tekstur bersifat homogen, sebaliknya *lacunarity* yang tinggi menunjukkan bahwa tekstur bersifat heterogen (Backes, 2013; Kilic dan Abiyev, 2011). Nilai *lacunarity* yang tinggi berarti bahwa piksel menyebar dalam rentang yang luas dan dikelilingi oleh *gap* yang banyak dan berukuran besar (Kilic dan Abiyev, 2011).

Awalnya, *lacunarity* muncul untuk mendeskripsikan karakteristik fraktal yang mempunyai dimensi sama namun memiliki penampakan berbeda (Dong, 2000; Backes, 2013). Hingga saat ini konsep *lacunarity* terus dikembangkan dalam menganalisis tekstur dan bersifat *scale-dependant* (Plotnick dkk., 1996; Dong, 2009; Backes, 2013; Mynt dan Lam, 2005). Metode yang paling sering digunakan dalam menghitung *lacunarity* dari suatu objek adalah metode *gliding box* (Allain dan Cloitre, 1991). Algoritma ini menganalisis citra dengan menerapkan kotak (*box*) yang bersifat *overlapping* dengan panjang  $r$ .



Gambar 2.4 Kotak berukuran  $r$  yang diterapkan pada citra biner 12x12 piksel (Plotnick dkk., 1993).

Plotnick dkk. (1993) mendemonstrasikan algoritma *gliding box* seperti berikut. Gambar 2.4 menunjukkan citra biner dengan ukuran  $12 \times 12$  piksel. Sebuah *box* dengan ukuran  $r \times r$  ( $r=2$ ) kemudian diletakkan di sebelah kiri atas

citra. Pada empat piksel dalam area kotak tersebut, terlihat bahwa dua piksel bernilai 1 berada dalam area kotak. Jumlah piksel yang berada dalam kotak disebut dengan *box mass* (massa kotak). Kotak kemudian bergerak satu kolom ke sebelah kanan dan massa kotak dihitung lagi. Proses ini kemudian berulang ke seluruh baris dan kolom hingga menghasilkan distribusi frekuensi dari massa kotak (Tabel 2.1). Jumlah kotak dengan ukuran  $r$  yang memiliki  $S$  piksel bernilai 1 ditandai dengan  $n(S,r)$  dan total jumlah kotak dengan ukuran  $r$  ditandai dengan  $N(r)$ . Jika ukuran citra adalah  $M$ , maka:

$$N(r) = (M - r + 1)^2, \quad (2.2)$$

Distribusi frekuensi ini kemudian dikonversi ke dalam distribusi probabilitas  $Q(S,r)$  dengan membagi distribusi frekuensi tersebut terhadap jumlah kotak

$$Q(S,r) = n(S,r)/N(r), \quad (2.3)$$

Momen pertama dan kedua dari distribusi kemudian ditentukan dengan

$$Z^{(1)} = \sum S Q(S,r), \quad (2.4)$$

$$Z^{(2)} = \sum S^2 Q(S,r), \quad (2.5)$$

Sehingga nilai *lacunarity*  $\Lambda$  dari citra dengan ukuran kotak  $r = 2$  bisa didefinisikan dengan

$$\Lambda(r) = Z^{(2)}/(Z^{(1)})^2. \quad (2.6)$$

Tabel 2.1 Perhitungan Nilai *Lacunarity* Dari Citra Masukan

$r=2$	S	$n(S,r)$	$Q(S,r)$	$SQ(S,r)$	$S^2Q(S,r)$
	0	3	0.024	0	0
	1	35	0.289	0.289	0.289
	2	46	0.380	0.760	1.520
	3	29	0.239	0.719	2.157
	4	4	0.066	0.264	1.057
			$Z^{(1)}=2.033$	$Z^{(2)}=5.024$	$\Lambda(r) = 1.215$

Sumber: Plotnick dkk. (1993).

Pada Tabel 2.1, terlihat bahwa nilai *lacunarity*  $\Lambda(2)$  dari citra adalah 1,215. Maka akan diperoleh sifat statistik dari  $\Lambda(2)$  dengan mengetahui bahwa:

$$Z^{(1)} = \bar{S}(r), \quad (2.7)$$

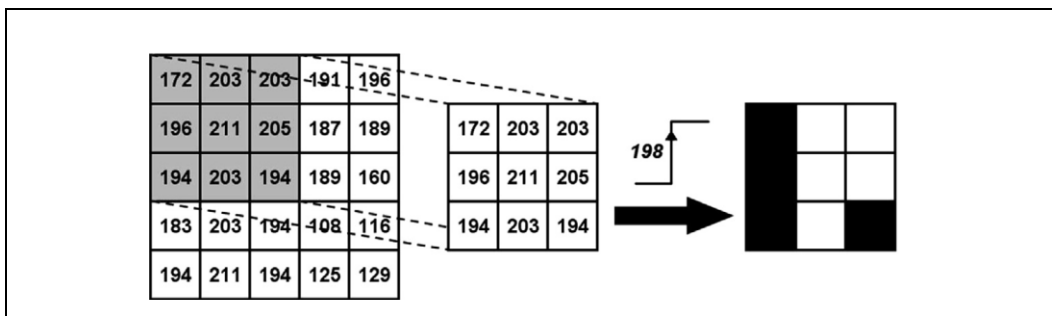
$$Z^{(2)} = s_s^2(r) + \bar{S}^2(r), \quad (2.8)$$

dengan  $\bar{S}(r)$  adalah rata-rata dan  $s_s^2(r)$  adalah varians dari jumlah piksel bernilai 1 dalam tiap kotak. Sebagai hasilnya, nilai *lacunarity*  $\Lambda$  didapatkan melalui formula

$$\Lambda(r) = s_s^2(r)/\bar{S}^2(r) + 1. \quad (2.9)$$

*Lacunarity* awalnya diajukan untuk analisis pada citra biner. Dengan adanya metode *thresholding* secara global, *lacunarity* dikembangkan agar bisa menganalisis citra tekstur *grayscale*. Namun demikian, tekstur umumnya merupakan karakteristik lokal pada citra. *Global thresholding* bisa menghilangkan informasi pada tekstur, sehingga mengurangi efisiensi dari *lacunarity* dalam mendeksripsikan pola dari tekstur.

Untuk menghindari kemungkinan hilangnya informasi pada saat perhitungan *lacunarity*, Backes (2013) mengajukan penggunaan *thresholding* lokal. Adapun proses *thresholding* dilakukan sebelum perhitungan distribusi massa  $n(S, r)$  untuk setiap posisi kotak dalam citra. Setiap piksel akan dianggap sebagai piksel *foreground* jika level keabuannya lebih tinggi daripada rata-rata level keabuan  $t$  dari keseluruhan piksel dalam kotak. Gambar 2.5 menunjukkan proses konversi citra *grayscale* dalam area kotak  $r = 3$ .



Gambar 2.5 Proses konversi citra *grayscale* dengan *thresholding* lokal. (Backes, 2013).

Jika  $T$  adalah tekstur lokal yang ada pada kotak dengan ukuran  $r \times r$ ,

$$T = \{g_1, g_2, \dots, g_{r^2}\}, \quad (2.10)$$



dimana nilai  $g_1 \in [0,255]$  merupakan nilai keabuan pada piksel lokal, maka massa  $m$  di dalam kotak dihitung dengan

$$m = \sum_{i=1}^{r^2} \delta(g_1), \quad (2.11)$$

dengan

$$\delta(g_1) = \begin{cases} 1, & g_i \geq t \\ 0, & g_i < t \end{cases} \quad (2.12)$$

Massa  $m$  ini kemudian dihitung pada setiap kotak  $r$  yang diterapkan pada citra, sehingga mempengaruhi distribusi massa  $n(S, r)$  yang merupakan masukan dalam perhitungan *lacunarity*. Selain melalui perhitungan rata-rata nilai keabuan, Backes (2013) menyarankan bahwa performa dari metode ini bisa lebih ditingkatkan dengan menggunakan teknik yang lebih berkembang seperti Otsu (1979) atau pengembangan adaptif.

## 2.3 Deteksi Tepi

Deteksi tepi berfungsi untuk memperoleh tepi objek. Deteksi tepi memanfaatkan perubahan nilai intensitas yang drastis pada batas dua area. Definisi tepi di sini adalah “himpunan piksel yang terhubung, yang terletak pada batas dua area” (Gonzalez & Woods, 2002). Tepi mengandung informasi yang sangat penting yang dapat berupa bentuk maupun ukuran dari suatu objek.

### 2.3.1 Operator Deteksi Tepi *Canny*

Algoritma deteksi tepi *Canny* (Canny, 1986) terdiri atas 5 langkah yang utama, yaitu:

1. *Smoothing*: Merupakan tahap penapisan terhadap citra dengan tujuan untuk menghilangkan derau. Operator deteksi tepi *Canny* menggunakan filter Gaussian yang dikonvolusi dengan citra masukan. Filter ini mengaburkan citra pada level yang ditentukan oleh sebuah nilai penentu ukuran filter, yaitu sigma  $\sigma$ . Tujuan penerapan filter Gaussian adalah untuk meminimalisir dampak dari munculnya informasi yang tidak diinginkan (*noise*). Semakin besar nilai  $\sigma$ , maka semakin kabur (*blurr*) penampakan dari citra yang akan berdampak pada semakin sedikitnya tepi yang bisa terdeteksi.

2. *Finding gradien* : Tepian harus ditandai pada gambar memiliki gradien yang besar.
3. *Non-maximum-suppresion* : Hanya maxima lokal yang harus ditandai sebagai wilayah tepi.
4. *Double thresholding* : *Thresholding* yang diterapkan untuk mendeteksi tepi yang potensial dengan menggunakan dua nilai ambang.
5. *Edge Tracking by hysteresis* : Tepian akhir ditentukan dengan menghilangkan semua tepi yang tidak tersambung dengan tepian yang sangat kuat (*strong edges*).

### 2.3.2 Operator Deteksi Tepi *Kirsch*

Operator gradien kompas adalah jenis operator yang mencari tepi dengan menggunakan delapan arah mata angin. Salah satu filter yang menerapkan gradient kompas adalah filter *kirsch* (Kirsch, 1971). Filter ini mengkonvolusi citra masukan dengan menggunakan 8 filter. Hasil operasinya berupa nilai maksimum dari ke delapan konvolusi. 8 *mask* dari filter *kirsch* terdiri atas *mask* berikut:

```
Barat Laut = [5 -3 -3; 5 0 -3; 5 -3 -3]
Barat Daya = [-3 -3 -3; 5 0 -3; 5 5 -3]
Tenggara   = [-3 -3 -3; -3 0 -3; 5 5 5]
Timur Laur  = [-3 -3 -3; -3 0 5; -3 5 5]
Utara       = [-3 -3 5; -3 0 5; -3 -3 -3]
Barat       = [-3 5 5; -3 0 5; -3 -3 -3]
Selatan     = [5 5 5; -3 0 -3; -3 -3 -3]
Timur       = [5 5 -3; 5 0 -3; -3 -3 -3]
```

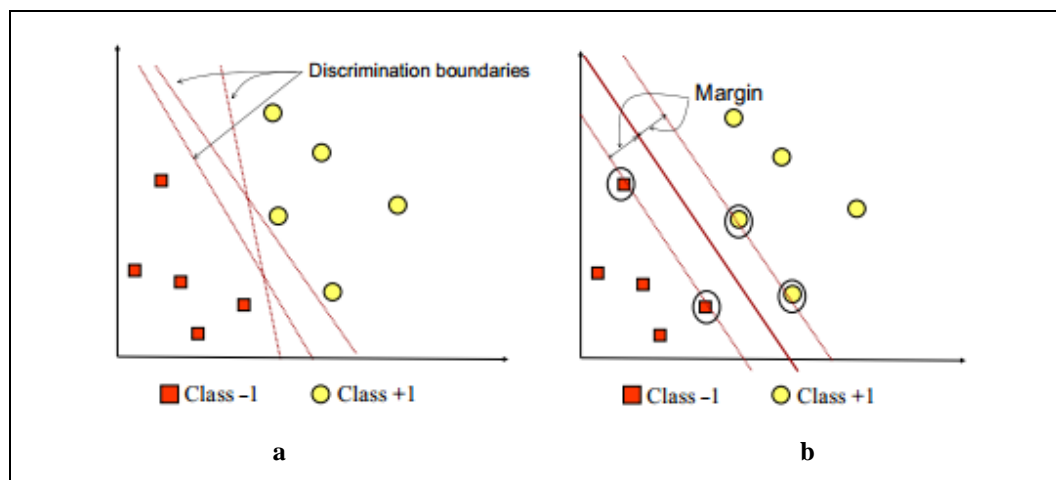
### 2.4 *k-fold Cross Validation System*

*Cross Validation* merupakan salah satu teknik untuk menilai/memvalidasi keakuratan sebuah model yang dibangun berdasarkan dataset tertentu. Pembuatan model biasanya bertujuan untuk melakukan prediksi maupun klasifikasi terhadap suatu data baru yang boleh jadi belum pernah muncul di dalam dataset. Pada *k-fold cross validation*, kadang juga disebut dengan estimasi rotasi, dataset *D* secara acak dibagi ke dalam *k* buah subset berukuran sama yang disebut dengan *fold D1*, *D2*, ... *Dk*. (Kohavi, 1995). Kemudian dilakukan sejumlah *k*-kali eksperimen,

dimana masing-masing eksperimen menggunakan satu data partisi ke- $k$  sebagai data uji dan memanfaatkan sisa partisi lainnya sebagai data latih. *10-cross validation* merupakan metode yang paling umum digunakan. Namun demikian, belum ada ketentuan baku mengenai nilai  $k$  yang paling tepat. Untuk mendapatkan nilai akurasi ataupun ukuran penilaian lainnya dari hasil eksperimen yang dilakukan, dapat diambil nilai rata-rata dari seluruh eksperimen tersebut.

## 2.5 Support Vector Machine

*SVM (Support Vector Machine)* diperkenalkan oleh Vapnik (1995) dan hingga kini digunakan pada banyak bidang penelitian seperti pengenalan pola, bioinformatika, diagnosa kanker, dan lain sebagainya (Li dkk., 2012). Menurut Nugroho dkk. (2003), *SVM* merupakan metode *machine learning* yang bertujuan menemukan *hyperplane* terbaik yang memisahkan dua buah kelas pada *input space*. Cara menemukan *hyperplane* ditunjukkan pada Gambar 2.6.



Gambar 2.6 Penentuan garis pemisah (a); dan garis pemisah terbaik (b). (Nugroho dkk., 2003).

Gambar 2.6 menunjukkan beberapa *pattern* yang merupakan anggota dari dua buah kelas : +1 dan -1. *Pattern* yang tergabung pada kelas -1 disimbolkan dengan warna merah (kotak), sedangkan *pattern* pada kelas +1, disimbolkan dengan warna kuning (lingkaran). Problem klasifikasi dapat diterjemahkan dengan usaha menemukan garis (*hyperplane*) yang memisahkan antara kedua

kelompok tersebut. Berbagai alternatif garis pemisah (*discrimination boundaries*) ditunjukkan pada Gambar 2.6 (a). *Hyperplane* pemisah terbaik antara kedua kelas dapat ditemukan dengan mengukur *margin hyperplane* tersebut dan mencari titik maksimalnya. *Margin* adalah jarak antara *hyperplane* tersebut dengan *pattern* terdekat dari masing-masing kelas. *Pattern* yang paling dekat ini disebut sebagai *support vector*. Garis tebal pada Gambar 2.6 (b) menunjukkan *hyperplane* yang terbaik, yaitu yang terletak tepat pada tengah-tengah kedua kelas, sedangkan titik merah dan kuning yang berada dalam lingkaran hitam adalah *support vector*. Pada SVM, terdapat parameter  $C$  berperan sebagai parameter *penalty*, yang menentukan seberapa besar nilai *penalty* yang akan kita berikan terhadap kesalahan klasifikasi.

Saat pertama kali diperkenalkan, SVM hanya dapat mengklasifikasikan data ke dalam dua kelas (klasifikasi biner). Ada beberapa jenis kernel SVM, dengan tipe kernel yang paling umum digunakan adalah kernel linear, dengan fungsi kernel:

$$K(x_i, x_j) = x_i^T x_j \quad (2.13)$$

dengan  $x_i$  adalah vektor data latih dan  $x_j$  adalah label kelas dari data latih, dengan parameter *penalty*  $C > 0$ .

Penelitian lebih lanjut untuk mengembangkan SVM sehingga bisa mengklasifikasi data yang memiliki lebih dari dua kelas terus dilakukan. Salah satu metode untuk mengimplementasikan SVM *multi-class* adalah metode *one against all*. Pada metode ini, dibangun  $k$  buah model SVM biner ( $k$  adalah jumlah kelas). Setiap model klasifikasi ke- $i$  dilatih dengan label positif, dengan menggunakan keseluruhan data yang diberi label negatif.

## BAB 3

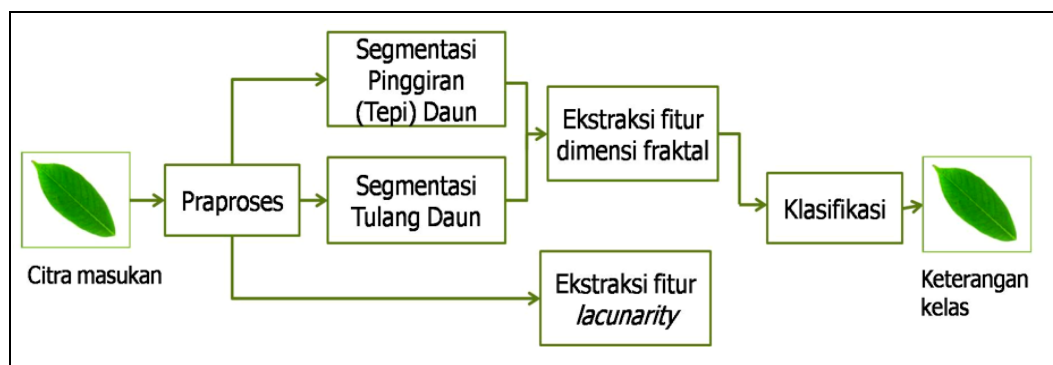
### METODOLOGI PENELITIAN

Bab ini memaparkan metodologi penelitian yang akan digunakan dalam merancang sistem untuk melakukan klasifikasi daun melalui penggabungan fitur dimensi fraktal dan *lacunarity*.

#### 3.1 Analisis dan Perancangan Sistem

Tahap analisis dalam penelitian ini meliputi perumusan masalah, penentuan batasan-batasan masalah, dan penyelesaiannya. Permasalahan yang akan diselesaikan adalah bagaimana mengekstraksi fitur bentuk dan tekstur pada citra daun sehingga diperoleh fitur dimensi fraktal dan *lacunarity* yang bisa dikombinasikan satu sama lain untuk meningkatkan akurasi klasifikasi. Setelah tahap analisis, kemudian dilakukan proses perancangan sistem klasifikasi daun yang menggabungkan fitur bentuk dimensi fraktal dan fitur tekstur *lacunarity* berdasarkan hasil analisis sebelumnya.

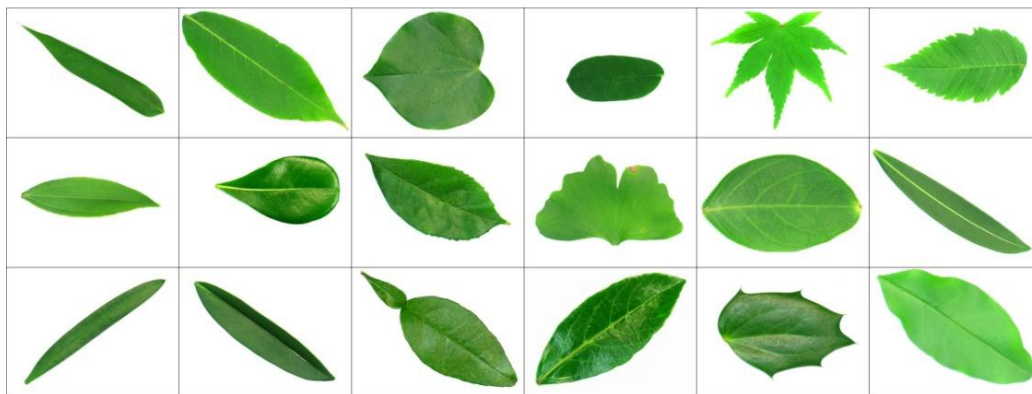
Tujuan dilakukannya proses perancangan dalam sebuah sistem adalah agar sistem dapat diimplementasikan sesuai dengan tujuan penelitian. Rancangan sistem menjelaskan proses atau tahapan apa saja yang berlangsung pada sistem. Secara garis besar, Rancangan sistem klasifikasi daun yang diajukan dalam penelitian ini terdiri atas empat proses, yaitu pengumpulan dataset, praproses dan segmentasi, ekstraksi fitur, dan klasifikasi. Gambar 3.1 menunjukkan rancangan sistem yang diajukan dalam penelitian ini.



Gambar 3.1 Diagram rancangan sistem secara umum.

### 3.1.1 Deskripsi Dataset

Pada penelitian ini, *dataset* yang digunakan adalah *dataset* daun dari *flavia* yang tersedia untuk umum. Gambar. 3.2 menunjukkan contoh *dataset* daun dari *flavia*. Citra dengan latar belakang putih diakuisisi dengan menggunakan *scanner*. Citra tersebut berukuran 1600×1200 piksel dengan format *JPEG*. *Dataset* dapat diunduh di situs <http://flavia.sourceforge.net/>.



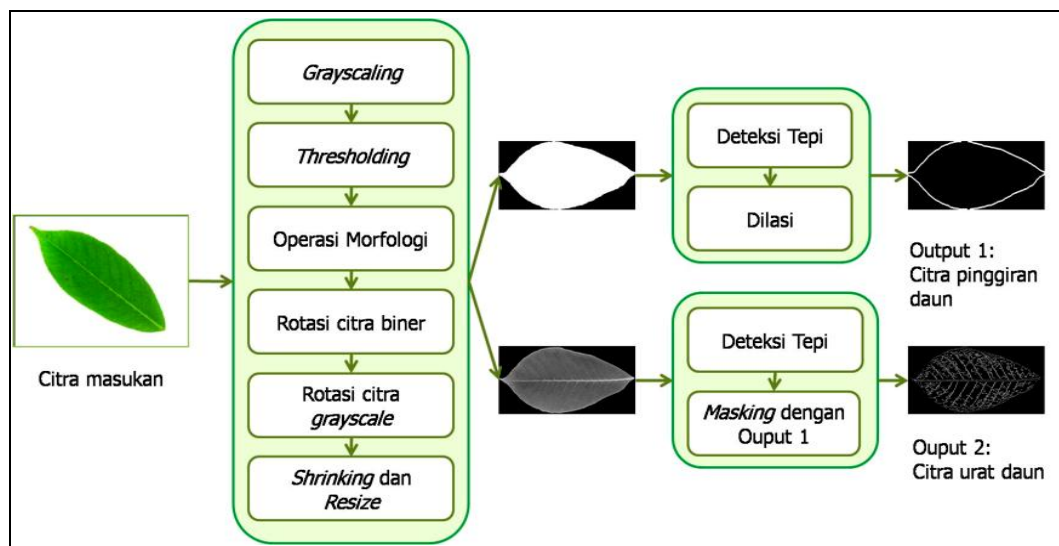
Gambar 3.2 Contoh *dataset* dari *flavia*.

### 3.1.2 Praproses dan Segmentasi

Sebelum suatu citra mengalami proses lebih lanjut, perlu dilakukan tahap praproses (*preprocessing*), yaitu teknik yang digunakan untuk mempersiapkan citra agar dapat menghasilkan citra keluaran yang diinginkan. Citra hasil praproses diharapkan dapat digunakan secara maksimal pada tahap selanjutnya. Pada penelitian ini, citra hasil praproses digunakan dalam proses segmentasi pinggiran daun dan urat daun. Citra hasil segmentasi merupakan citra yang menjadi citra masukan pada proses ekstraksi fitur.

Pada proses analisis bentuk dari pinggiran dan tekstur daun, terlebih dahulu dilakukan tahap praproses agar nantinya dihasilkan fitur yang bersifat *translation* dan *rotation invariant*. Fitur bersifat *translation invariant* berarti bahwa fitur dimensi fraktal yang dihasilkan akan bernilai sama atau tidak mengalami perubahan yang sangat signifikan jika citra mengalami pergeseran dengan besar tertentu. Begitu pula pada variasi arah perputaran citra. Fitur akan disebut *rotation invariant* jika nilai yang dihasilkan tidak mengalami perubahan

signifikan jika citra masukan mengalami perputaran yang beragam. Awalnya, citra dikonversi ke citra *grayscale* dan biner. Citra biner tidak sepenuhnya tersegmentasi dengan benar sehingga dilakukan proses dilasi dan erosi untuk mengisi area non-objek pada citra, dengan menerapkan *structuring element* “disk” dan *radius* = 15. Agar fitur yang dihasilkan nantinya juga bersifat *invariant* terhadap rotasi, maka dilakukan proses perhitungan nilai panjang dari *major axis* dari citra biner (Chaki dkk., 2015). Kemudian dihitung nilai orientasi dari citra atau besar sudut  $\theta$  antara *major axis* terhadap sumbu  $x$  (sumbu horizontal). Besar sudut rotasi citra adalah sama dengan sudut  $\theta$  yang diperoleh. Agar fitur yang dihasilkan nantinya juga bersifat *invariant* terhadap translasi, maka dilakukan proses *shrinking*, dimana citra otomatis akan di-*crop* sesuai dengan ukuran maksimum citra. Proses *shrinking* diterapkan pada citra biner dan citra *grayscale* yang telah dirotasi. Gambar 3.3 adalah diagram tahapan dalam memperoleh pinggiran daun dan urat daun dengan menggabungkan metode oleh Chaki dkk. (2015) dan Bruno dkk. (2008).



Gambar 3.3 Praproses dan segmentasi pinggiran dan urat daun.

Pada Gambar 3.3 terdapat proses *resize* yang dilakukan untuk meperkecil kemungkinan pengaruh dari proses rotasi dan *shrinking*. Proses rotasi dan *shrinking* pada tahap sebelumnya akan menghasilkan citra keluaran yang

mempunyai ukuran bervariasi. Ukuran citra yang tidak sama akan berdampak pada panjang fitur yang tidak seragam, sehingga diperlukan tahap *resize* untuk menyeragamkan panjang atau lebar dari citra yang akan menjadi masukan pada tahap selanjutnya. Oleh karena itu, setelah tahap *shrinking*, citra di-*resize* sesuai dengan *aspect ratio* dari citra, yaitu perbandingan antara panjang dan lebar citra (Chaki, 2015). Proses *resize* dilakukan dengan menerapkan aturan yang dapat dilihat pada Tabel 3.1. Untuk memperoleh *aspect ratio*  $R$  dari citra daun maka digunakan persamaan:

$$R = \frac{\text{panjangcitra}}{\text{lebarcitra}} \quad (3.1)$$

Keluaran dari proses ini adalah citra ukuran  $m \times n$  sesuai dengan kriteria aspek rasio  $R$  yang ditetapkan. Tujuan penetapan ukuran dengan kriteria *aspect ratio* ini adalah agar dihasilkan fitur vektor dengan panjang yang seragam dan bersifat *scale invariant*. Fitur yang bersifat *scale invariant* adalah fitur yang akan bernilai sama atau tidak mengalami perubahan yang signifikan walaupun citra masukan mengalami perubahan skala dalam besaran tertentu.

Tabel 3.1 Aspek Rasio  $R$  dengan Ukuran Citra Berbeda.

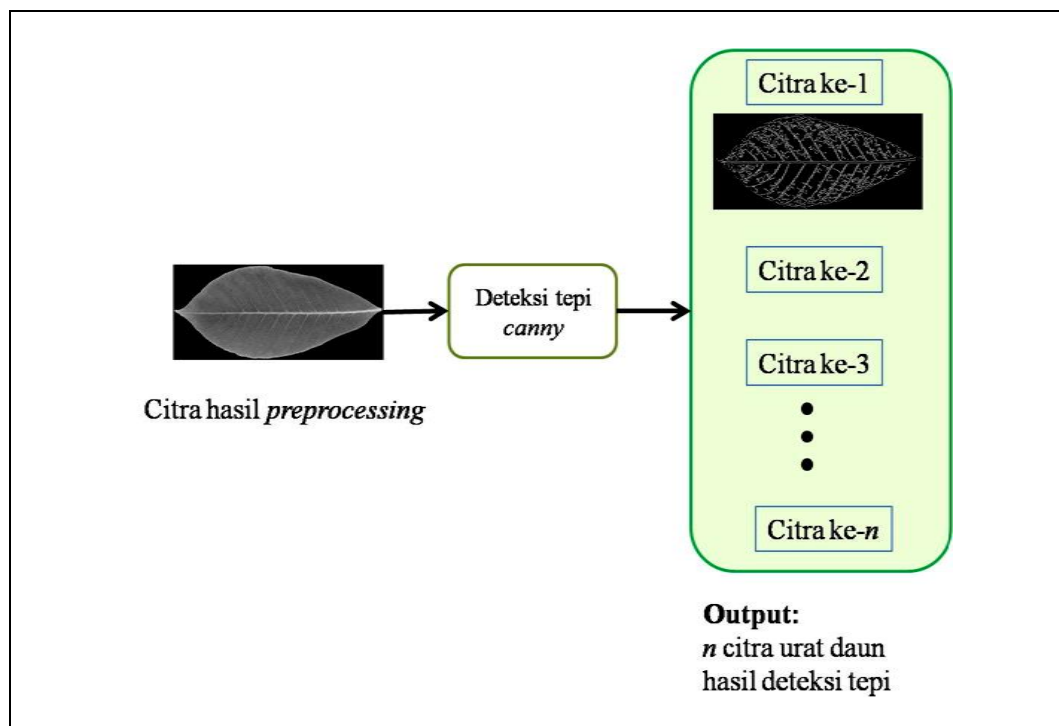
<i>Aspect Ratio R</i>	Ukuran citra ( $m \times n$ )
$R \leq 1.4$	450 x 450
$1.4 < R \leq 2$	300 x 450
$2 < R \leq 2.4$	210 x 450
$2.4 < R \leq 3$	150 x 450
$3 < R \leq 5$	98 x 450
$5 < R \leq 13$	68 x 450
$R > 13$	15 x 450

Untuk memperoleh citra bentuk urat/tulang daun, terlebih dahulu dilakukan praproses dengan menerapkan tahapan yang sama dengan proses deteksi pinggiran daun (*leaf contour*). Mendeteksi urat daun dan memisahkannya dari daun merupakan proses yang cukup rumit karena perbedaan kontras yang

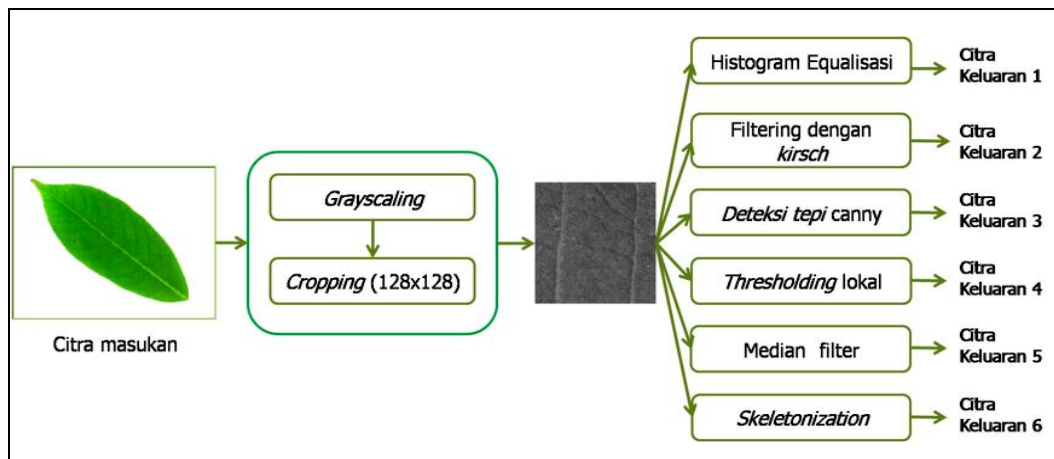


sangat rendah antara urat daun dan objek daun (Du dkk., 2013; Bruno dkk., 2008). Oleh karena itu, Bruno dkk. (2008) menerapkan satu nilai ambang untuk melakukan segmentasi urat daun, seperti terlihat pada Gambar 3.3. Operasi dilasi diterapkan pada citra pinggiran daun dengan menggunakan *structuring element* “disk” dan *radius* = 2. Citra hasil dilasi ini akan dijadikan *mask* pada citra hasil segmentasi urat daun.

Adapun Du dkk. (2013) menerapkan metode segmentasi yang bersifat *multi-thresholding* dalam memisahkan urat daun sehingga diperoleh lebih dari satu citra urat daun seperti terlihat pada Gambar 3.4. Pada penelitian ini digunakan penggabungan metode dari kedua penelitian tersebut, dimana untuk memperoleh urat daun akan diterapkan ide dari Du dkk. (2013), yang menggunakan lebih dari satu citra urat daun dengan nilai ambang berbeda, sedangkan proses *masking* dilakukan dengan menerapkan metode dari Bruno dkk. (2008).



Gambar 3.4 Proses segmentasi urat daun dengan  $n$  nilai ambang.

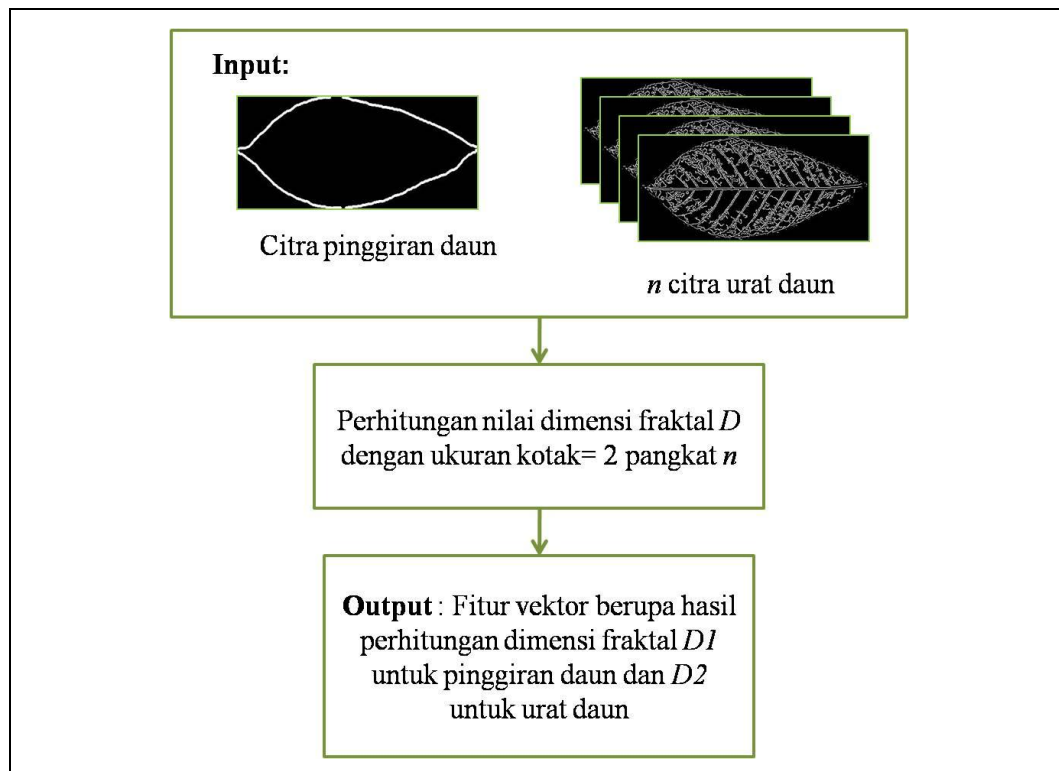


Gambar 3.5 Tahapan praproses citra tekstur.

Gambar 3.5 merupakan tahapan praproses untuk memperoleh citra tekstur dari daun. Awalnya, dilakukan proses konversi citra RGB ke citra *grayscale*, dilanjutkan dengan tahapan *cropping* sehingga dihasilkan citra berukuran 128x128 piksel. Selanjutnya dilakukan proses ekualisasi histogram pada citra *grayscale* dengan tujuan untuk meratakan kontras citra *grayscale* sehingga detail dari tekstur daun menjadi lebih jelas. Citra hasil histogram merupakan citra keluaran pertama pada tahap praproses ini. Selanjutnya diterapkan *filter kirsch* pada citra *grayscale*. Alasan pemilihan filter ini adalah karena kemampuannya dalam memperjelas tepi pada citra *grayscale* melalui analisis pada 8 arah mata angin. Citra hasil *filtering* dengan metode *kirsch* ini merupakan citra keluaran ke-2 pada tahap praproses. Citra ke-3 yang menjadi keluaran adalah citra biner hasil *local thresholding* (Backes, 2013). Citra keluaran ke-4 adalah citra hasil deteksi tepi *canny*. Alasan pemilihan ini adalah karena tekstur juga memiliki tepi yang menunjukkan informasi spasial dari citra. Citra yang menjadi keluaran ke-5 adalah citra hasil proses *skeletonization* atau penulangan. Proses ini diterapkan untuk memperoleh informasi tekstur dalam bentuk satu piksel garis tanpa menghilangkan informasi penting dari citra. Citra keluaran terakhir adalah citra hasil *median filtering* dari citra *local thresholding*.

### 3.1.3 Ekstraksi Fitur Dimensi Fraktal dan *Lacunarity*

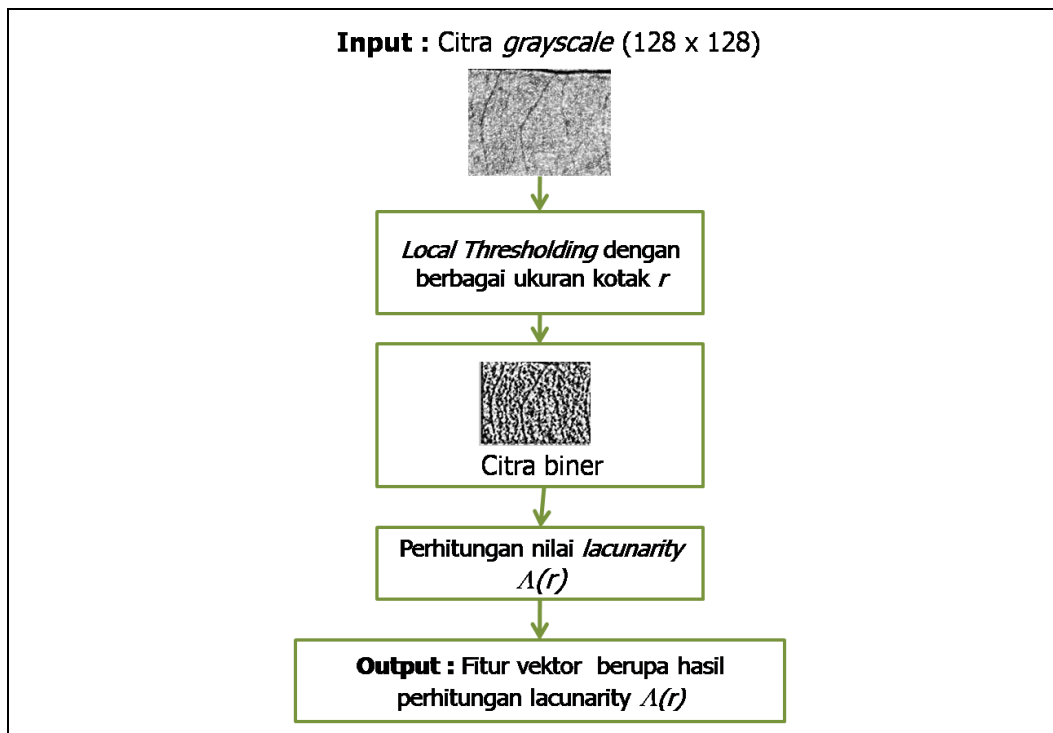
Untuk memperoleh fitur dimensi fraktal dan *lacunarity* dari daun, dilakukan dua kali proses ekstraksi fitur. Pertama, untuk memperoleh fitur dimensi fraktal, bentuk dari citra pinggiran daun dan urat daun dianalisis dengan menggunakan metode perhitungan dimensi fraktal yaitu *box counting* (Bruno dkk., 2008). Metode ini membandingkan hasil log banyaknya jumlah kotak  $N(r)$  terhadap banyaknya kotak ukuran  $r$  yang dibutuhkan untuk menutupi seluruh citra. Selanjutnya diperoleh fitur nilai dimensi fraktal  $D1$  untuk citra pinggiran daun. Sedangkan nilai dimensi fraktal  $D2$  dari urat daun terdiri atas  $n$  fitur vektor citra hasil segmentasi dengan nilai sigma  $\sigma$  berbeda pada operator deteksi tepi. Gambar 3.7 menunjukkan proses ekstraksi fitur dimensi fraktal dari bentuk pinggiran daun dan bentuk tulang/urat daun.



Gambar 3.6 Diagram ekstraksi fitur dimensi fraktal dari pinggiran dan urat daun.

Untuk memperoleh fitur *lacunarity* dari tekstur daun, citra tekstur daun dianalisis dengan menggunakan metode *gliding box* (Plotnick, 1993; Backes,

2013). Pada tahap ini, kotak dengan ukuran  $r$  bergerak pada citra *grayscale* dan biner dimulai pada bagian kiri atas citra. Sebelum menghitung nilai *lacunarity*, dilakukan *local thresholding*. Kemudian dilakukan perhitungan jumlah piksel bernilai 0 yang ada di dalam kotak. Setelah kotak selesai diterapkan, maka distribusi frekuensi dari *massa* kotak dihitung sehingga nilai *lacunarity* bisa didapatkan melalui persamaan 2.6. Pada penelitian ini, ukuran kotak  $r$  adalah parameter yang akan dianalisis dalam skenario uji coba. Gambar 3.7 menunjukkan diagram proses ekstraksi fitur tekstur dari satu citra biner hasil *local thresholding* dengan menerapkan metode oleh Backes (2013) untuk memperoleh fitur vektor *lacunarity*.



Gambar 3.7 Diagram ekstraksi fitur *lacunarity* dari tekstur daun.

Penggabungan fitur dilakukan dengan menambahkan satu vektor fitur dengan vektor fitur lainnya. Oleh karena itu, fitur vektor dimensi fraktal dari bentuk daun digabungkan dengan fitur vektor *lacunarity* dari tekstur daun, sehingga menghasilkan sebuah fitur vektor sepanjang  $1 \times n$ , dimana  $n$  terdiri atas kombinasi antara fitur  $D1$ ,  $D2$  dan  $\Lambda r$ . Adapun  $D1$  adalah dimensi fraktal dari

bentuk pinggiran daun,  $D_2$  adalah dimensi fraktal dari bentuk urat daun, dan  $Ar$  adalah *lacunarity* dari tekstur daun pada kotak dengan ukuran  $r$ . Panjang fitur akan dipengaruhi oleh banyaknya kotak  $r$  yang diterapkan serta jumlah citra masukan.

### 3.1.4 Klasifikasi

Pada tahap klasifikasi, *dataset* akan dibagi ke dalam 10 partisi dan klasifikasi dilakukan dengan menggunakan pengklasifikasi *Support Vector Machine (SVM)*. Meskipun pada awalnya *SVM* merupakan pengklasifikasi untuk kasus dua kelas saja, pengembangan *SVM*, yaitu *Support Vector Machine* yang bersifat multi kelas, dapat melakukan klasifikasi ke banyak kelas melalui metode *one against all*. Alasan pemilihan pengklasifikasi ini adalah karena kemampuannya dalam generalisasi, implementasinya yang relatif mudah, serta kemampuannya dalam menangani data berdimensi tinggi (Nugroho dkk., 2003). Adapun tipe kernel yang digunakan adalah tipe kernel linear dengan nilai parameter *penalty C* adalah 0.5.

### 3.2 Implementasi

Pada tahap ini, dilakukan implementasi terhadap desain yang telah dirancang sebelumnya. Implementasi dilakukan menggunakan perangkat lunak Matlab R2011b dengan memanfaatkan *library* yang sudah tersedia.

### 3.3 Uji Coba dan Evaluasi

Pengujian performa dari pengklasifikasi dilakukan dengan menerapkan metode *k-fold cross validation* (Kohavi, 1995). Pada penelitian ini, pengujian klasifikasi daun dilakukan dengan menggunakan nilai  $k=10$ . Sebagai gambaran, jika dilakukan *10-Fold Cross-Validation* maka data eksperimen  $D$  akan dibagi ke dalam 10 subset yang dinamai sebagai data  $D_1, D_2, D_3, \dots, D_{10}$ . Desain data pengujian dengan menggunakan *10-Fold Cross-Validation* dari 10 subset data yang telah terbentuk dapat dilihat pada tabel 3.2.

Tabel 3.2 Desain Data Pengujian.

Nomor <i>Fold</i>	Data Latih	Data Uji
1	D2,D3,D4,D5,D6,D7,D8,D9,D10	D1
2	D1,D3,D4,D5,D6,D7,D8,D9,D10	D2
3	D1,D2,D4,D5,D6,D7,D8,D9,D10	D3
4	D1,D2,D3,D5,D6,D7,D8,D9,D10	D4
5	D1,D2,D3,D4,D6,D7,D8,D9,D10	D5
6	D1,D2,D2,D4,D5,D7,D8,D9,D10	D6
7	D1,D2,D3,D4,D5,D6,D8,D9,D10	D7
8	D1,D2,D3,D4,D5,D6,D7,D9,D10	D8
9	D1,D2,D3,D4,D5,D6,D7,D8,D10	D9
10	D1,D2,D3,D4,D5,D6,D7,D8,D9	D10

Semua *dataset* dibagi ke dalam 10 “*fold*” subset data. Satu subset digunakan sebagai data uji sedangkan 9 subset lainnya digunakan sebagai data latih. Prosedur ini berulang untuk masing-masing subset. Untuk mendapatkan nilai akurasi ataupun ukuran penilaian lainnya dari hasil pengujian yang dilakukan, maka diambil nilai rata-rata dari seluruh pengujian tersebut.

$$Akurasi (\%) = \frac{jumlah\ data\ yang\ terklasifikasi\ benar}{jumlah\ keseluruhan\ data} \times 100$$

Adapun skenario uji coba pada penelitian ini adalah dengan membandingkan hasil klasifikasi ke dalam beberapa macam kondisi, yaitu:

#### 1. Skenario Uji Coba ke-1

Uji coba ke-1 merupakan uji coba dengan hanya menggunakan fitur dimensi fraktal dari pinggiran dan urat daun. Pada skenario ini, banyaknya jumlah citra urat daun yang diperoleh dengan menerapkan operator deteksi tepi *canny* juga dianalisis untuk mengetahui apakah hal tersebut akan berdampak pada hasil klasifikasi. Selain itu, dilakukan pula percobaan untuk melihat pengaruh proses rotasi, *shrinking* dan *resize* terhadap akurasi klasifikasi.

#### 2. Skenario Uji Coba ke-2

Uji coba ke-2 merupakan uji coba dengan hanya menggunakan fitur *lacunarity*. Tujuan uji coba ini adalah untuk membuktikan adanya pengaruh ukuran kotak  $r$  pada metode *gliding box lacunarity* terhadap akurasi klasifikasi. Pada percobaan ini juga diamati pengaruh banyaknya citra inputan terhadap akurasi klasifikasi. Percobaan ini juga akan membandingkan dengan metode yang sudah ada sebelumnya.

### 3. Skenario Uji Coba ke-3

Uji coba ke-3 dilakukan dengan menerapkan usulan metode untuk menggabungkan fitur dimensi fraktal dan *lacunarity*. Pada tahap ini dilakukan penggabungan fitur vektor hasil perhitungan dimensi fraktal dengan metode *box counting* pada uji coba ke-1 dengan fitur vektor hasil perhitungan *lacunarity* dengan metode *gliding box* hasil uji coba ke-2. Dari kombinasi ini diharapkan diperoleh nilai hasil klasifikasi yang lebih baik dari dua percobaan sebelumnya.

### 3.4 Analisis Hasil

Pada tahap ini dilakukan analisis dari hasil uji coba fitur dimensi fraktal, uji coba fitur *lacunarity*, serta hasil uji coba dari penggabungan fitur dimensi fraktal dan *lacunarity*. Selain itu, dilakukan pula analisis parameter ukuran kotak (*box*)  $r$  yang paling optimal dalam menghasilkan akurasi terbaik pada perhitungan *lacunarity* serta analisis penyebab kesalahan klasifikasi. Diharapkan dari analisis uji coba ini akan diperoleh hasil yang sesuai dengan tujuan penelitian.

## BAB 4

### HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas tentang hasil uji coba dari penelitian yang telah dilakukan. Berdasarkan dari hasil uji coba maka akan dilakukan proses analisis untuk mengetahui beberapa hal yang akan menjadi kesimpulan dari penelitian ini.

#### 4.1 Lingkungan Uji Coba

Untuk uji coba pada penelitian ini, spesifikasi dari perangkat keras yang digunakan dalam implementasi perangkat lunak terdiri atas prosesor berjenis Intel Core i3 2.20 GHz, memori berkapasitas 2.00 GB dan *harddisk* dengan kapasitas sebesar 500 GB. Sedangkan untuk spesifikasi perangkat lunak yang digunakan dalam implementasi perangkat lunak yakni Microsoft Excel, ImageJ, serta Matlab R2011b.

#### 4.2 Data Uji Coba

Tabel 4.1 menunjukkan nama spesies dan jumlah per-kelas dari *dataset*.

Tabel 4.1 Keterangan *Dataset*.

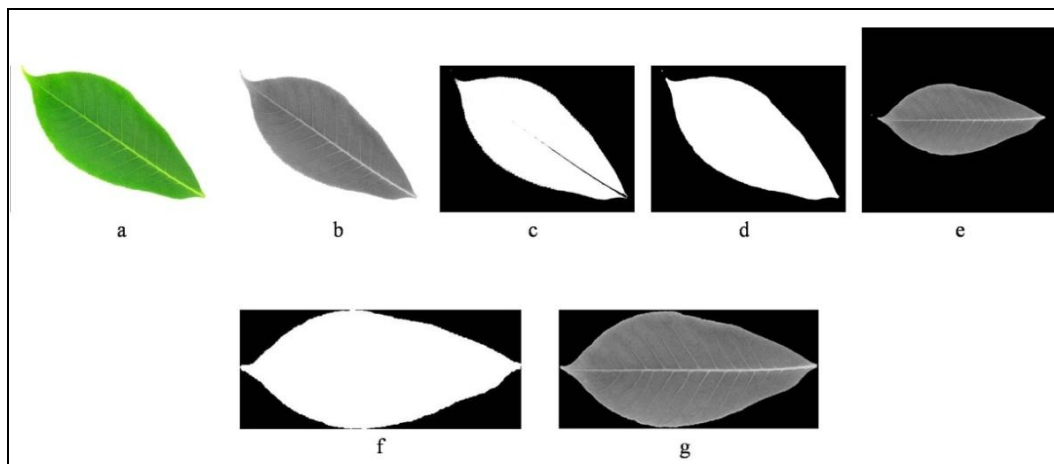
Label Kelas	Nama Spesies	Jumlah data per-kelas
1	<i>pubescent bamboo</i>	34
2	<i>Chinese horse chestnut</i>	36
3	<i>Chinese redbud</i>	37
4	<i>true indigo</i>	48
5	<i>Japanese maple</i>	37
6	<i>goldenrain tree</i>	29
7	<i>Chinese cinnamon</i>	37
8	<i>Japanese cheesewood</i>	36
9	<i>Sweet osmanthus</i>	29
10	<i>Ginkgo</i>	36
11	<i>Crepe myrtle</i>	39
12	<i>Oleander</i>	33
13	<i>yew plum pine</i>	32
14	<i>Ford Woodlotus</i>	27
15	<i>Tangerine</i>	26
16	<i>Japan Arrowwood</i>	36
17	<i>Beales Barberry</i>	39
18	<i>Glossy Privet</i>	35



Data uji coba yang dijelaskan pada Tabel 4.1 merupakan data yang digunakan dalam penelitian ini. Data ini bersumber dari dataset daun *flavia*. Adapun total jumlah dataset yang dijadikan sebagai data uji coba adalah sebanyak 626 citra, yang terdiri atas daun dari 18 kelas/spesies yang berbeda. Jumlah data per-kelas bervariasi antara satu dengan yang lainnya.

### 4.3 Praproses

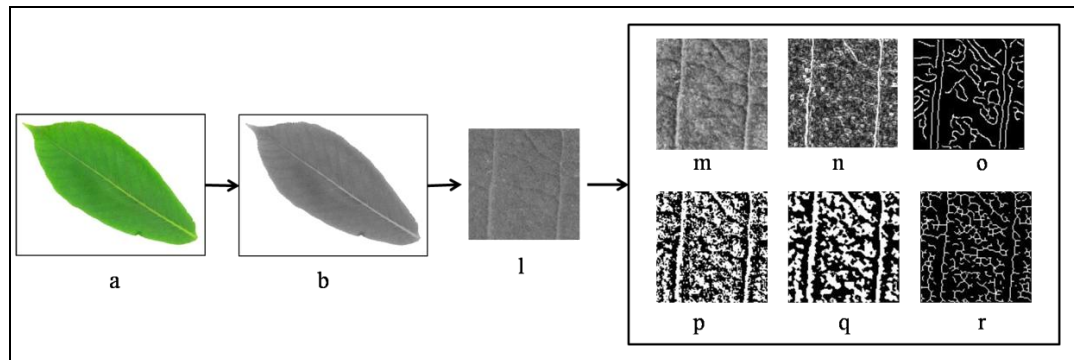
Pada penelitian ini, terdapat beberapa tahapan yang dilakukan untuk memperoleh citra keluaran berupa bentuk dan tekstur daun yang diharapkan. Gambar 4.1 menunjukkan hasil dari tahap praproses untuk memperoleh citra yang akan digunakan pada tahap segmentasi pinggiran dan urat daun. Adapun contoh citra yang digunakan adalah salah satu citra daun dari kelas 2 (*Chinese horse chestnut*).



Gambar 4.1 Tahap praproses citra. (a): citra asli; (b): citra *grayscale*; (c): citra biner; (d): citra hasil dilasi dan erosi; (e): citra hasil rotasi; (f): citra biner hasil *shrinking* dan *resize*; (g): citra *grayscale* hasil *shrinking* dan *resize*.

Pada Gambar 4.1, citra masukan adalah citra RGB berukuran  $1600 \times 1200$  piksel (a). Citra ini kemudian dikonversi ke citra *grayscale* dan biner (b dan c). Citra biner tidak sepenuhnya tersegmentasi sehingga dilakukan proses dilasi dan erosi (d), kemudian dilanjutkan dengan tahap rotasi (e) agar citra bersifat *rotation invariant*. Agar citra bersifat *invariant* terhadap translasi, maka dilakukan proses *shrinking* dan *resize* pada citra biner dan citra *grayscale* yang telah dirotasi (f dan

g). Hasil akhir dari tahap praproses adalah terbentuknya satu citra biner dan satu citra *grayscale* yang akan menjadi masukan pada tahap ekstraksi fitur dimensi fraktal.



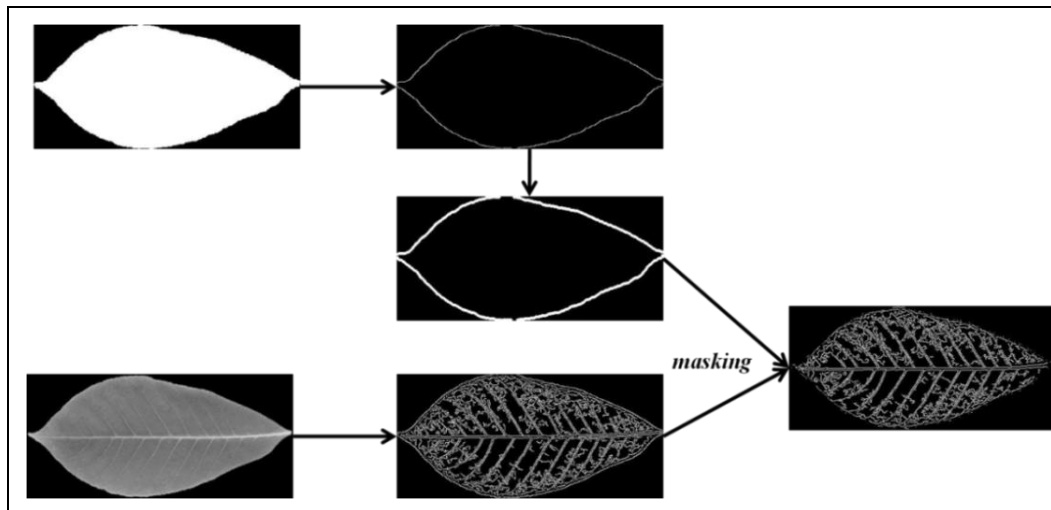
Gambar 4.2 Tahap praproses untuk mendapatkan citra tekstur daun.

Gambar 4.2 merupakan tahapan praproses untuk memperoleh tekstur dari daun. Awalnya, dilakukan proses konversi citra RGB ke citra *grayscale* (b), dilanjutkan dengan tahapan *cropping* sehingga menghasilkan citra berukuran  $128 \times 128$  piksel (l). Selanjutnya dilakukan proses histogram equalisasi (m), penerapan filter *kirsch* pada citra *grayscale* (n), operasi deteksi tepi dengan metode *canny* (o), operasi *thresholding* lokal (p) serta penerapan *median filtering* dari citra hasil *thresholding* lokal (q), dan diakhiri dengan proses *skeletonization* (r). Hasil akhir dari tahap praproses ini adalah terbentuknya enam citra keluaran yang akan menjadi citra masukan pada tahap ekstraksi fitur *lacunarity*.

#### 4.4 Segmentasi Pinggiran dan Urat Daun

Pinggiran atau *contour* dari citra daun diperoleh dengan menerapkan operator deteksi tepi *canny* pada citra biner. Untuk memperoleh *mask* dari pinggiran daun, maka dilakukan proses dilasi pada citra pinggiran daun. Citra urat daun juga diperoleh melalui penerapan operator deteksi tepi *canny* pada citra *grayscale*. Selanjutnya citra dilakukan proses *masking* citra urat daun dengan menggunakan citra pinggiran daun. Hasil proses segmentasi adalah citra pinggiran daun hasil dilasi dan citra urat daun hasil *masking* (citra urat daun tanpa

pinggiran). Gambar 4.3 merupakan hasil proses segmentasi pinggiran daun dan urat daun yang diajukan dalam penelitian ini.



Gambar 4.3 Proses segmentasi pinggiran dan urat daun.

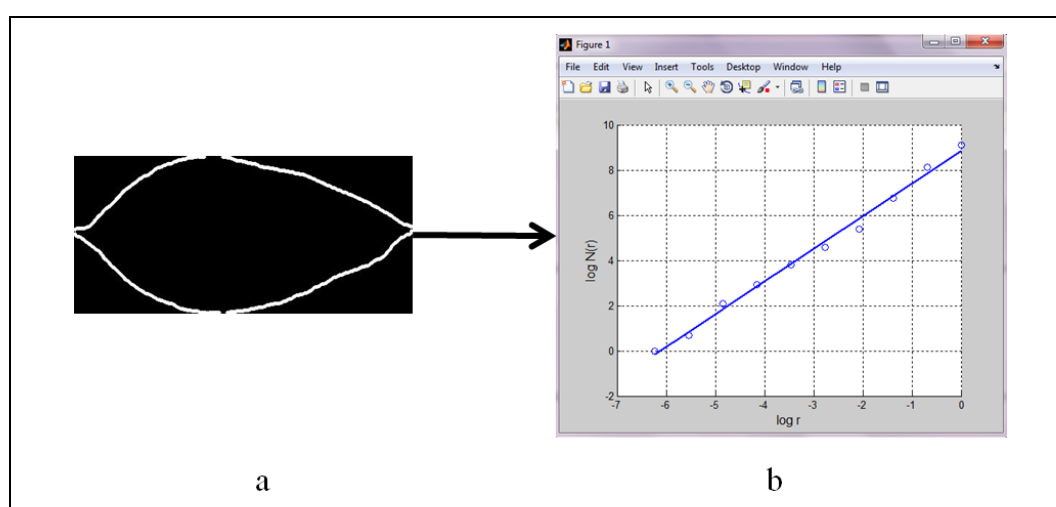
## 4.5 Ekstraksi Fitur

Berikut ini adalah hasil proses ekstraksi fitur dari citra daun, dengan mengambil salah satu contoh daun spesies *Chinese horse chestnut* dari kelas 2. Tahap ekstraksi fitur terdiri atas dua proses utama, yaitu perhitungan nilai dimensi fraktal dan perhitungan nilai *lacunarity*. Perhitungan nilai dimensi fraktal dengan metode *box counting* diterapkan pada citra pinggiran daun dan citra urat daun hasil segmentasi. Sedangkan perhitungan nilai *lacunarity* dengan metode *gliding box* diterapkan pada citra tekstur daun hasil *preprocessing*.

### 4.5.1 Ekstraksi Fitur Dimensi Fraktal

Untuk memperoleh fitur dimensi fraktal dari citra pinggiran daun, maka diterapkan metode *box counting* pada citra masukan. Tabel 4.2 menunjukkan proses perhitungan nilai fitur dimensi fraktal dari pinggiran daun, yang diperoleh dari grafik perbandingan  $\log$  banyaknya kotak  $N(r)$  terhadap  $\log$  ukuran kotak  $r$ . Berdasarkan grafik, diperoleh koordinat nilai-nilai  $x$  dan  $y$  yang cocok (*fit*) terhadap persamaan garis lurus pada kurva perbandingan  $\log N(r)$  terhadap  $r$ . Nilai kemiringan (*slope*) atau  $m$  dari persamaan garis lurus  $y=mx+b$  kemudian dihitung. Nilai *slope* ini merupakan nilai dimensi fraktal dari citra masukan,

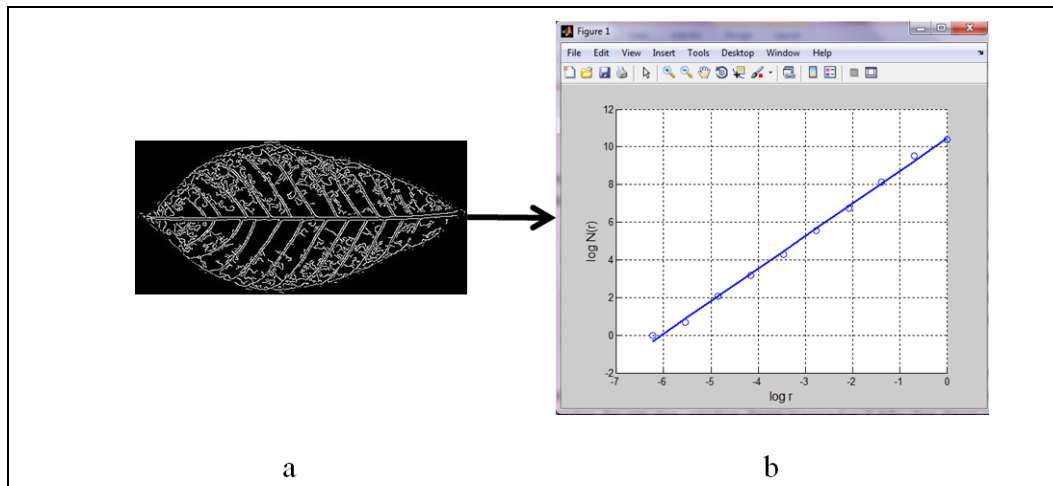
sedangkan *diff value* (nilai perbedaan) merupakan nilai hasil perbandingan  $\log N(r)$  terhadap  $r$  pada sepanjang garis lurus yang membentuk sebuah vektor sepanjang  $1 \times 9$ . Sembilan nilai ini kemudian digabungkan dengan nilai *slope* (dimensi fraktal), sehingga diperoleh fitur vektor akhir *D1* sepanjang  $1 \times 10$  untuk satu citra pinggiran daun. Perlakuan yang sama juga diberikan pada citra urat daun yang menjadi masukan kedua pada proses perhitungan dimensi fraktal, seperti terlihat pada Tabel 4.3. Fitur vektor *D2* sepanjang  $1 \times 10$  juga dihasilkan dari proses perhitungan dimensi fraktal dari citra urat daun ini.



Gambar 4.4 Proses penentuan dimensi fraktal dari pinggiran daun. (a): citra pinggiran daun; (b): *log-log plot* dari hasil perbandingan  $N(r)$  dan  $r$ .

Tabel 4.2 Contoh Proses Perhitungan Nilai Fitur Dimensi Fraktal dari Pinggiran Daun Melalui Grafik dengan Metode *Box Counting*.

Fitur	Ukuran kotak $r$									
	512	256	128	64	32	16	8	4	2	1
$N(r)$	1	2	8	19	44	95	214	541	1522	4813
$\log r$	-6.238	-5.545	-4.852	-4.159	-3.466	-2.773	-2.079	-1.386	-0.693	0.000
$\log N(r)$	0.000	0.693	2.079	2.944	3.807	4.595	5.398	6.755	8.118	9.094
<i>bestFit</i>	-0.161	0.841	1.843	2.845	3.847	4.849	5.851	6.854	7.856	8.858
$\text{diff}_{\log(r)}$	-0.693	-0.693	9.011	-0.693	-0.693	-0.693	-0.693	2.079	-0.693	
$\text{diff}_{\log N(r)}$	-0.693	-1.386	-0.865	-0.862	-0.788	-0.803	-1.356	-1.363	-0.977	
$\text{diff } N(r) : \text{diff } r$	1.000	2.000	-0.096	1.244	1.138	1.159	1.957	-0.655	1.409	
<i>Slope</i>	1.4457									
<b>Fitur Akhir</b>	1.4457	1.000	2.000	-0.096	1.244	1.138	1.159	1.957	-0.655	1.409



Gambar 4.5 Proses penentuan dimensi fraktal dari urat daun. (a): citra urat daun; (b): *log-log plot* dari hasil perbandingan  $N(r)$  terhadap  $r$ .

Tabel 4.3 Contoh Proses Perhitungan Nilai Fitur Dimensi Fraktal dari Urat Daun Melalui Grafik dengan Metode *Box Counting*.

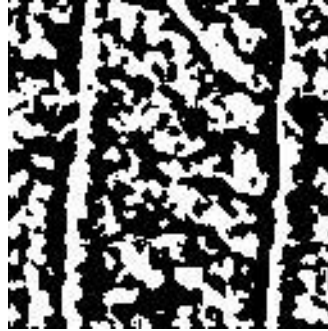
Fitur	Ukuran kotak $r$									
	512	256	128	64	32	16	8	4	2	1
$N(r)$	1	2.00	8.00	24	74	257	857	3426	13616	32533
$\log r$	-6.238	-5.545	-4.852	-4.159	-3.466	-2.773	-2.079	-1.386	-0.693	0.000
$\log N(r)$	0.000	0.693	2.079	3.178	4.304	5.549	6.753	8.139	9.519	10.390
<i>bestFit</i>	-0.327	0.870	2.067	3.265	4.462	5.659	6.857	8.054	9.251	10.449
<i>diff log r</i>	-0.693	-0.693	-0.693	-0.693	-0.693	-0.693	-0.693	-0.693	-0.693	
<i>diff log Nr</i>	-0.693	-1.386	-1.099	-1.126	-1.245	-1.204	-1.386	-1.380	-0.871	
<i>diff N(r) : diff r</i>	1.000	2.000	1.585	1.624	1.796	1.738	1.999	1.991	1.257	
<i>slope</i>	1.727									
<b>Fitur Akhir</b>	1.727	1.000	2.000	1.585	1.624	1.796	1.738	1.999	1.991	1.257

Fitur vektor  $DI$  sepanjang  $1 \times 10$  hasil perhitungan dimensi fraktal dari citra pinggiran daun kemudian digabungkan dengan fitur vektor  $D2$  sepanjang  $1 \times 10$  hasil perhitungan dimensi fraktal citra urat daun. Gabungan fitur ini dianggap mampu merepresentasikan keunikan bentuk dari citra daun yang menjadi contoh. Adapun fitur vektor  $D$  sepanjang  $1 \times 20$  hasil penggabungan fitur dimensi fraktal dari pinggiran dan urat daun adalah sebagai berikut:

$$\overline{D} = [1.4457 \quad 1.000 \quad 2.000 \quad -0.096 \quad 1.244 \quad 1.138 \quad 1.159 \quad 1.957 \quad -0.655 \quad \dots \\ 1.409 \quad 1.724 \quad 1.000 \quad 2.000 \quad 1.585 \quad 1.624 \quad 1.796 \quad 1.738 \quad 1.999 \quad 1.991 \quad 1.257]$$

#### 4.5.2 Ekstraksi Fitur *Lacunarity*

Metode *gliding box* diterapkan untuk memperoleh fitur vektor *lacunarity* dari citra masukan. Pada metode ini, beragam ukuran kotak  $r$  diterapkan pada citra tekstur daun, dengan  $r = 2^n$ . Citra tekstur daun hasil praproses memiliki ukuran sebesar  $128 \times 128$  piksel, sehingga maksimum ukuran kotak yang bisa diterapkan pada citra adalah sebesar  $r = 128$ . Gambar 4.6 merupakan contoh salah satu dari enam citra masukan pada tahap ini.



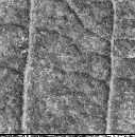
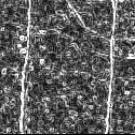



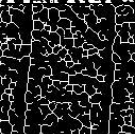
Gambar 4.6 Contoh citra hasil *local thresholding* (Backes, 2013).

Pada Gambar 4.6, ukuran kotak  $r$  yang diterapkan pada citra adalah  $r = 2, 4, 8, 16, 32, 64$  dan  $128$ . Untuk menghitung nilai *lacunarity*, awalnya kotak yang bersifat *overlap* dengan ukuran  $r = 2 \times 2$  bergerak dari kiri ke kanan citra. Jumlah piksel yang ada dalam setiap kotak  $r$  kemudian dihitung sehingga menghasilkan distribusi frekuensi. Pada citra yang menjadi contoh, total jumlah kotak yang diperlukan adalah sebanyak 16129 kotak. Sehingga diperoleh total akhir jumlah piksel yang berada jika  $r = 2$  adalah 7078124. Selanjutnya dilakukan perhitungan nilai kuadrat dari hasil perhitungan pertama, sehingga diperoleh nilai sebesar  $5.9956 \times 10^9$ .

*Lacunarity* diperoleh dengan persamaan yang telah dijelaskan pada Bab 2, sehingga pada citra biner ini diperoleh nilai *lacunarity* sebesar 1.9302. Perlakuan yang sama diberikan kepada 5 citra tekstur yang menjadi citra masukan dalam proses perhitungan *lacunarity*. Tabel 4.4 merupakan hasil proses ekstraksi fitur

*lacunarity* dengan menggunakan metode *gliding box* dari salah satu citra kelas 2 (*Chinese horse chestnut*).  $Z^{(1)}$  adalah momen pertama dari distribusi frekuensi peluang kemunculan  $S$  dengan menerapkan ukuran kotak  $r$ , yang dapat diperoleh melalui persamaan 2.4, sedangkan  $Z^{(2)}$  adalah momen kedua dari distribusi yang diperoleh melalui persamaan 2.5. Nilai *lacunarity*  $\Lambda(r)$  diperoleh melalui persamaan 2.6. Ukuran fitur vektor yang terbentuk dari proses ini adalah  $1 \times 6$ , yang terdiri atas nilai *lacunarity* dari masing-masing citra masukan.

Tabel 4.4 Hasil Perhitungan Nilai Lacunarity  $\Lambda r$  dari 6 Citra Masukan.

Citra masukan	$Z^{(1)}$	$Z^{(2)}$	$\Lambda(r)$
	7887918	$4.0078 \times 10^9$	1.0389
	6063061	$2.8126 \times 10^9$	1.1955
	2040150	$1.0299 \times 10^9$	3.9909
	7392341	$5.6676 \times 10^9$	1.6728
	7078124	$5.9956 \times 10^9$	1.9302
	2325928	$1.0835 \times 10^9$	3.2302

#### 4.6 Uji Coba

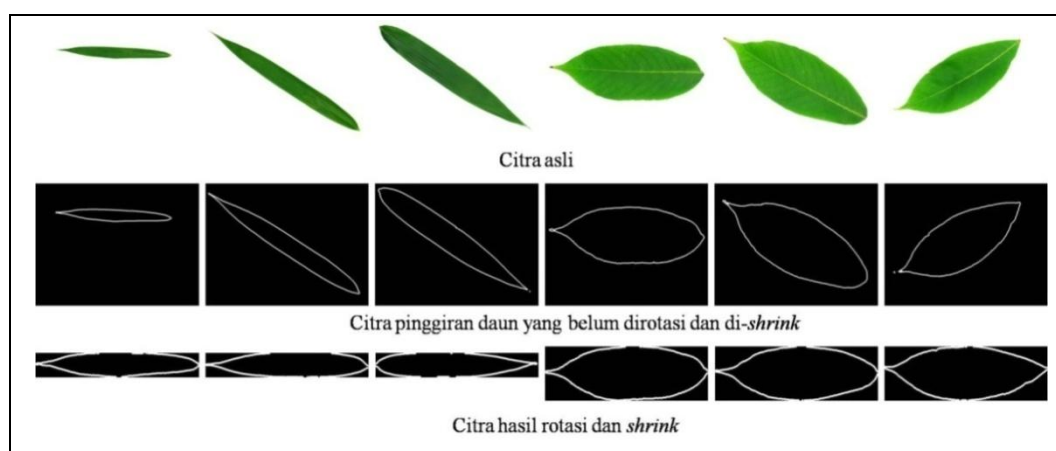
Pada penelitian ini dilakukan beberapa uji coba sesuai dengan rancangan yang telah ditentukan pada Bab 3. Uji coba ini bertujuan untuk membuktikan hipotesis bahwa ada sinergi antara dimensi fraktal pada analisis bentuk dengan *lacunarity* pada analisis tekstur pada kasus klasifikasi daun. Semua percobaan

dilakukan dengan menerapkan pengklasifikasi *Support Vector Machine (SVM)* yang bersifat multi-kelas. Adapun fungsi kernel yang digunakan adalah kernel linear dengan nilai dari parameter *penalty C* adalah 0.5. Performa dari sistem divalidasi dengan menggunakan *k-cross validation system* dengan  $k = 10$ . Untuk memperoleh nilai akurasi klasifikasi, dilakukan perhitungan nilai rata-rata dari hasil klasifikasi dari tiap *fold*.

#### 4.6.1 Hasil Uji Coba Fitur Dimensi Fraktal

##### a. Pengaruh Proses Rotasi dan *Shrinking*

Percobaan pertama bertujuan untuk membuktikan adanya pengaruh proses rotasi dan *shrinking* terhadap akurasi klasifikasi menggunakan fitur dimensi fraktal. Dengan menerapkan metode *box counting* pada citra pinggiran dan urat daun, dilakukan percobaan dengan menggunakan 107 citra daun dari 3 kelas berbeda untuk mengetahui apakah variasi dalam translasi dan rotasi dapat mempengaruhi akurasi. Gambar 4.7 menunjukkan beberapa contoh citra dengan arah rotasi yang berbeda. Tabel 4.5 menunjukkan hasil klasifikasi pada uji coba ini. Hasil uji coba menunjukkan nilai akurasi sebesar 87% untuk *dataset* yang tidak mengalami proses rotasi dan *shrinking* serta 89.64% untuk citra yang mengalami proses rotasi dan *shrinking*. Hal ini menunjukkan bahwa dengan melakukan proses rotasi, *shrinking*, serta *resize*, maka hasil klasifikasi menjadi lebih stabil karena tidak adanya variasi dalam rotasi maupun translasi.



Gambar 4.7 Proses rotasi dan *shrinking* citra.



Tabel 4.5 Perbandingan Hasil Klasifikasi Antara Citra yang Mengalami Rotasi dan *Shrinking* Dengan Citra Tanpa Rotasi dan *Shrinking*.

Nomor <i>Fold</i>	Akurasi (%)	
	Tanpa rotasi dan <i>shrink</i>	Dengan rotasi dan <i>shrink</i>
1	90.00	90.00
2	100.00	100.00
3	81.82	90.91
4	90.91	90.91
5	100.00	90.91
6	63.64	81.82
7	81.82	90.91
8	81.82	90.91
9	80.00	80.00
10	100.00	90.00
<b>Rata-rata (%)</b>	<b>87.00</b>	<b>89.64</b>

#### b. Pengaruh Jumlah Nilai Sigma $\sigma$ pada Operator *Canny*

Percobaan selanjutnya dalam analisis fitur dimensi fraktal dengan metode *box counting* adalah penerapan operator deteksi tepi *canny* (Canny J, 1976). Operator *canny* diterapkan untuk memperoleh citra bentuk pinggiran daun dan citra bentuk urat daun. Percobaan ini dilakukan dengan menerapkan beberapa nilai sigma  $\sigma$  pada proses deteksi tepi citra urat daun. Percobaan ini bertujuan untuk mengetahui pengaruh dari jumlah nilai sigma  $\sigma$  pada operator *canny* terhadap hasil klasifikasi. Tabel 4.6 menunjukkan hasil perbandingan akurasi klasifikasi pada percobaan ini. Gambar 4.8 merupakan contoh proses deteksi tepi dengan menerapkan 4 nilai sigma  $\sigma$ , sehingga diperoleh 4 citra urat daun.

Tabel 4.6 Hasil Klasifikasi Dengan Menggunakan Beragam Nilai Sigma Pada Operator *Canny*.

Citra masukan	Panjang fitur vektor	Akurasi klasifikasi (%)
Pinggiran daun, Urat daun ( $\sigma = 1$ ) (Bruno, 2008)	1x20	60.376
Pinggiran daun, Urat daun ( $\sigma = 1$ and $\sigma = 2$ )	1x30	71.556
Pinggiran daun, Urat daun ( $\sigma = 1, \sigma = 2, \sigma = 3$ )	1x40	73.458
Pinggiran daun, Urat daun ( $\sigma = 1, \sigma = 2, \sigma = 3, \sigma = 4$ )	1x50	<b>76.979</b>

## BAB 5

### KESIMPULAN DAN SARAN

Pada bab ini dijelaskan mengenai kesimpulan yang diperoleh setelah melakukan serangkaian uji coba dan analisis, serta saran untuk pengembangan penelitian lebih lanjut.

#### 5.1. Kesimpulan

Pada penelitian ini telah dilakukan analisis terhadap pengaruh penggabungan antara fitur dimensi fraktal dan *lacunarity* terhadap hasil klasifikasi daun. Eksperimen dilakukan dengan menggunakan 626 citra daun dari *flavia*. Fitur dimensi fraktal diperoleh dengan menerapkan metode *box counting* pada citra bentuk dari pinggiran dan urat daun, sedangkan fitur *lacunarity* diperoleh dari implementasi metode *gliding box* pada citra tekstur daun. Dari hasil analisis uji coba yang telah dilakukan, diperoleh kesimpulan sebagai berikut:

1. Dengan menggunakan hasil ekstraksi fitur dimensi fraktal dari citra pinggiran dan urat daun, diperoleh akurasi klasifikasi sebesar 76.979% ketika menerapkan gabungan nilai sigma  $\sigma=1, 2, 3$ , dan 4 pada operator deteksi tepi *canny* dan menggunakan fitur dari persamaan garis lurus pada kurva.
2. Ekstraksi fitur *lacunarity* dengan menggunakan metode sebelumnya menghasilkan akurasi klasifikasi yang rendah, yaitu 19.83%. Hal ini disebabkan karena satu citra biner tidak cukup untuk merepresentasikan fitur *lacunarity* dari tekstur daun. Penggunaan 6 citra masukan serta ukuran kotak  $r = 2, 4, 8, 16$  dan 32 menghasilkan sebuah fitur vektor *lacunarity* yang bisa meningkatkan akurasi klasifikasi menjadi 50.169%. Namun demikian, terjadi kesalahan klasifikasi yang disebabkan oleh tingginya kemiripan antar-kelas pada citra tekstur daun.
3. Gabungan antara fitur dimensi fraktal dari bentuk daun dan *lacunarity* dari tekstur daun menghasilkan fitur vektor yang dapat merepresentasikan citra daun sehingga akurasi klasifikasi bisa meningkat hingga 93.92%. Hal ini

menunjukkan bahwa terdapat sinergi positif antara fitur dimensi fraktal dan fitur *lacunarity*.

## 5.2 Saran

Berdasarkan hasil yang didapatkan dari penelitian ini, terdapat beberapa saran yang berguna untuk penelitian-penelitian selanjutnya, yaitu:

1. Perlu penelitian khusus mengenai metode segmentasi tulang atau urat daun, sebab mendeteksi tulang daun merupakan pekerjaan yang cukup sulit.
2. Akurasi klasifikasi pada fitur *lacunarity* masih rendah ketika bekerja sendiri, utamanya pada kasus banyak kelas dengan kemiripan yang tinggi. Ke depannya, fitur *lacunarity* lebih tepat jika dikombinasikan dengan fitur lain.

## DAFTAR PUSTAKA

- Allain, C., dan Cloitre, M. (1991). Characterizing the lacunarity of random and deterministic fractal sets. *Physical review A*, 44(6), 3552.
- Arun, C. H., Sam, E. W. R., dan Christopher, D.D. (2013). Texture feature extraction for identification of medicinal plants and comparison of different classifiers. *Int J Comput Appl*, 62(12), 1-9.
- Backes, A. R., Casanova, D., dan Bruno, O. M. (2009). Plant leaf identification based on volumetric fractal dimension. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(06), 1145-1160.
- Backes, A. R. (2013). A new approach to estimate lacunarity of texture images. *Pattern Recognition Letters*, 34(13), 1455-1461.
- Beghin, T., Cope, J. S., Remagnino, P., dan Barman, S. (2010). Shape and texture based plant leaf classification. In *Advanced Concepts for Intelligent Vision Systems* (pp. 345-353). Springer Berlin Heidelberg.
- Borys, P., Krasowska, M., Grzywna, Z. J., Djamgoz, M. B., dan Mycielska, M. E. (2008). Lacunarity as a novel measure of cancer cells behavior. *Biosystems*, 94(3), 276-281.
- Bruno, O. M., de Oliveira Plotze, R., Falvo, M., dan de Castro, M. (2008). Fractal dimension applied to plant identification. *Information Sciences*, 178(12), 2722-2733.
- Casanova, D., de Mesquita Sa Junior, J. J., dan Bruno, O. M. (2009). Plant leaf identification using Gabor wavelets. *International Journal of Imaging Systems and Technology*, 19(3), 236-243
- Canny, J. (1986). A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6), 679-698.
- Chaki, J., Parekh, R., dan Bhattacharya, S. (2015). Plant leaf recognition using texture and shape features with neural classifiers. *Pattern Recognition Letters*, 58, 61-68.
- Dirr, I. (1990). Manual of Woody Landscape Plants: Their Identification, Ornamental Characteristics, Culture, Propagation and Uses.

- Dong, P. (2000). Test of a new lacunarity estimation method for image texture analysis. *International Journal of Remote Sensing*, 21(17), 3369-3373.
- Dong, P. (2009). Lacunarity analysis of raster datasets and 1D, 2D, and 3D point patterns. *Computers & Geosciences*, 35(10), 2100-2110.
- Du, J. X., Wang, X. F., dan Zhang, G. J. (2007). Leaf shape based plant species recognition. *Applied mathematics and computation*, 185(2), 883-893.
- Du, J. X., Zhai, C. M., dan Wang, Q. P. (2013). Recognition of plant leaf image based on fractal dimension features. *Neurocomputing*, 116, 150-156.
- Gonzalez, RC dan Woods, RE (2002). Digital Image Processing. Upper Saddle River, New Jersey: Prentice Hall.
- Kadir, A., Nugroho, L. E., Susanto, A., dan Santosa, P. I. (2013). Leaf classification using shape, color, and texture features. *arXiv preprint arXiv:1401.4447*.
- Kaplan, D.R. (2001). The science of plant morphology: definition, history, and role in modern biology. *Am J Bot.* 88(10):1711-41
- Kilic, K. I., dan Abiyev, R. H. (2011). Exploiting the synergy between fractal dimension and lacunarity for improved texture recognition. *Signal Processing*, 91(10), 2332-2344.
- Kirsch, R. A. (1971). Computer determination of the constituent structure of biological images. *Computers and biomedical research*, 4(3), 315-328.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *IJCAI (Vol. 14, No. 2, pp. 1137-1145)*.
- Li, L., Chang, L., Ke, S., dan Huang, D. (2012). Multifractal analysis and lacunarity analysis: A promising method for the automated assessment of muskmelon (*Cucumis melo* L.) epidermis netting. *Computers and Electronics in Agriculture*, 88, 72-84.
- Lin, F. Y., Zheng, C. H., Wang, X. F., dan Man, Q. K. (2008). Multiple classification of plant leaves based on gabor transform and lbp operator. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Contemporary Intelligent Computing Techniques* (pp. 432-439). Springer Berlin Heidelberg.

- Mandelbrot, B. B. (1982). *The Fractal Geometry of Nature*. San Francisco, CA: Freeman.
- Myint, S. W., dan Lam, N. (2005). A study of lacunarity-based texture analysis approaches to improve urban image classification. *Computers, environment and urban systems*, 29(5), 501-523.
- Neves, L. A., Nascimento, M. Z., Oliveira, D. L. L., Martins, A. S., Godoy, M. F., Arruda, P. F. F., dan Machado, J. M. (2014). Multi-scale lacunarity as an alternative to quantify and diagnose the behavior of prostate cancer. *Expert Systems with Applications*, 41(11), 5017-5029.
- Novotný, P., dan Suk, T. (2013). Leaf recognition of woody species in Central Europe. *biosystems engineering*, 115(4), 444-452.
- Nugroho, A. S., Witarto, A. B., dan Handoko, D. (2003). Support Vector Machine Teori dan Aplikasinya dalam Bioinformatika. *Kuliah Umum IlmuKomputer. Com*.
- Otsu, N. (1979). A threshold selection method from gray level histograms. *IEEE Trans. Syst. Man Cybern.* 9, 62–66.
- Plotnick, R. E., Gardner, R. H., dan O'Neill, R. V. (1993). Lacunarity indices as measures of landscape texture. *Landscape ecology*, 8(3), 201-211.
- Plotnick, R. E., Gardner, R. H., Hargrove, W. W., Prestegard, K., dan Perlmutter, M. (1996). Lacunarity analysis: a general technique for the analysis of spatial patterns. *Physical review E*, 53(5), 5461.
- Putra, D. (2010). Pengolahan citra digital. Penerbit Andi Yogyakarta.
- Sosa, J. P. M., Solano, G. A., dan Obico, J. A. (2013) An Application Using Canny Edge Detection And Multilayer Perceptron for Recognizing Leaves Of Tropical Plants. *Mathematics And Computers In Biology And Biomedical Informatics*.
- Sembiring, K. (2007). Penerapan Teknik Support Vector Machine untuk Pendeteksian Intrusi pada Jaringan. S1 Teknik Informatika, Sekolah Teknik Elektro dan Informatika, ITB.
- Vapnik, V., dan Cortes, C. (1995). Support-vector networks, *Machine Learning*, 20(3), 273-297

- Voss, R. F. (1986). Characterization and measurement of random fractals. *Physica Scripta*, 1986(T13), 27.
- Xu, Y., Quan, Y., Ling, H., dan Ji, H. (2011). Dynamic texture classification using dynamic fractal analysis. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 1219-1226). IEEE.
- Dataset daun *flavia* : <http://flavia.sourceforge.net/>

## BIODATA PENULIS



**Mutmainnah Muchtar**, lahir di kota Kendari, Sulawesi Tenggara pada tanggal 12 Januari 1991, merupakan anak kedua dari empat bersaudara. Penulis menempuh pendidikan dari SDN Inpres Unhalu Kendari, SMPN 1 Kendari, dan SMAN 4 Kendari. Pada tahun 2008 penulis melanjutkan ke jenjang S1 Teknik Informatika di Universitas Haluoleo (UHO) Kendari. Kemudian pada tahun 2013 dengan rahmat Allah SWT,

penulis diterima di Magister Teknik Informatika Fakultas Teknologi Informasi ITS melalui jalur BPPDN (Beasiswa Pendidikan Pascasarjana Dalam Negeri) Dikti dan berhasil menyelesaikan studi pada tahun 2015 dengan bidang minat KCV (Komputasi Cerdas dan Visualisasi). Penulis memiliki ketertarikan dalam bidang *image processing*, Sistem biometrika, serta *intelligent transport system*. Penulis memiliki hobi membaca dan menggambar ini dapat dihubungi melalui email : [muchtarmutmainnah@gmail.com](mailto:muchtarmutmainnah@gmail.com).